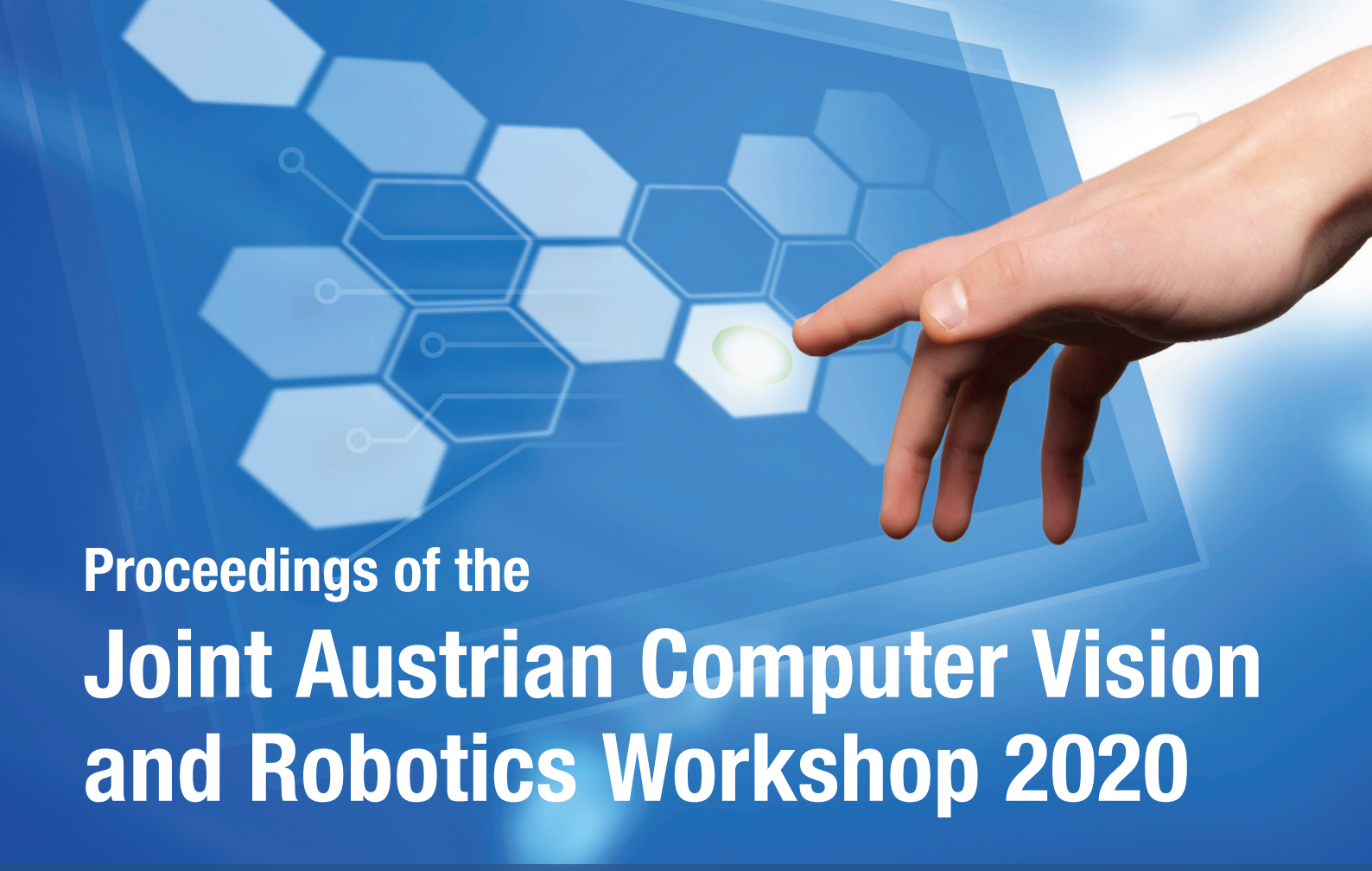


Proceedings of the

Joint Austrian Computer Vision and Robotics Workshop 2020

OAGM - Austrian Association for Pattern Recognition

GMAR - Gesellschaft für Mess-, Automatisierungs- und Robotertechnik

TU Graz
Graz University of Technology

OAGM

OESTERREICHISCHE COMPUTER GESELLSCHAFT
AUSTRIAN COMPUTER SCCIETY

GMAR
Die österreichische Gesellschaft für Mess-, Automatisierungs- und Robotertechnik

Peter M. Roth, Gerald Steinbauer, Friedrich Fraundorfer, Mathias Brandstötter, and Roland Perko (eds.)

Proceedings of the

Joint Austrian Computer Vision and Robotics Workshop 2020

Graz University of Technology

Graz, Austria

Austrian Association of Pattern Recognition (OAGM)

Gesellschaft für Mess-, Automatisierungs- und Robotertechnik (GMAR)

Editors

Peter M. Roth, Gerald Steinbauer, Friedrich Fraundorfer, Mathias Brandstötter, and Roland Perko

Layout

Austrian Association of Pattern Recognition
https://aapr.at/

Gesellschaft für Mess-, Automatisierungs- und Robotertechnik

http://www.gmar.at/

Cover

Verlag der Technischen Universität Graz

Supported by:









© 2020 Verlag der Technischen Universität Graz
www.tugraz-verlag.at

ISBN 978-3-85125-752-6
DOI 10.3217/978-3-85125-752-6



https://creativecommons.org/licenses/by/4.0/deed.en

Contents

Preface iv

Workshop Organization v

Program Committee vi

Awards 2019 viii

Index of Authors x

Austrian Robotics Workshop 1

Semi-Automatic Generation of Training Data for Neural Networks for 6D Pose Estimation and Robotic Grasping
Johannes Rauer, Mohamed Aburaia, and Wilfried Wöber 2

Feasibility study of a certifiable production environment using safe environmental sensor systems
Maximilian Papa, Vinzenz Sattinger, and Wilfried Kubinger 4

Vision-based Docking of a Mobile Robot
Andreas Kriegler and Wilfried Wöber 6

Autonomous Grasping of Known Objects Using Depth Data and the PCA
Dominik Steigl, Wilfried Wöber, and Mohamed Aburaia 13

EDLRIS: European Driving License for Robots and Intelligent Systems
Manuel Menzinger, Martin Kandlhofer, Gerald Steinbauer, Ronald Bieber, Wilfried Baumann, Margit Ehardt-Schmiederer, and Thomas Winkler 19

Design and Implementation of a Mobile Search and Rescue Robot
Georg Novotny and Wilfried Kubinger 21

Automatic Ontology-based Plan Generation for an Industrial Robotics System
Timon Hoebert, Wilfried Lepuschitz, and Munir Merdan 27

How does explicit exploration influence Deep Reinforcement Learning?
Jakob Hollenstein, Erwan Renaudo, Matteo Saveriano, and Justus Piater 29

UGV Radiation Mapping using a Particle Filter
Alexander Permann, Daniel Hettegger, and Gerald Steinbauer 31

Towards ASP-based Scheduling for Industrial Transport Vehicles
Felicitas Fabricius, Marco De Bortoli, Selmair Maximilian, Michael Reip, Gerald Steinbauer, and Martin Gebser 34

Learning Manipulation Tasks from Vision-based Teleoperation
Matthias Hirschmanner, Ali Jamadi, Bernhard Neuberger, Timothy Patten, and Markus Vincze 42

Reactive motion planning framework inspired by hybrid automata
Csaba Hajdu and Áron Ballagi 48

Automated Log Ordering through Robotic Grasper
Stephan Weiss, Stefan Ainetter, Fred Arneitz, Dailys Arronde, Rohit Dhakate, Friedrich Fraundorfer, Harald Gietler, Wolfgang Gubensäk, Mylena Medeiros, Christian Stetco, and Hubert Zangl 50

Introducing a Morphological Box for an Extended Risk Assessment of Human-Robot Work Systems Considering Prospective System Modifications
Titanilla Komenda, Martin Steiner, Michael Rathmair, and Mathias Brandstötter . . 53

Several Approaches for the Optimization of Arm Motions of Humanoids
Daniel Lichtenecker, Gabriel Krög, Hubert Gattringer, and Andreas Müller 59

OAGM Workshop 64

Presentation Attacks and Their Detection in Finger and Hand Vein Recognition
Luca Debiasi, Christof Kauba, Heinz Hofbauer, Bernhard Prommegger, and Andreas Uhl 65

HPS: Holistic End-to-End Panoptic Segmentation Network with Interrelations
Günther Kniewasser, Alexander Grabner, and Peter M. Roth 71

Frame-To-Frame Consistent Semantic Segmentation
Manuel Rebol and Patrick Knöbelreiter 79

Ground Control Point Retrieval From SAR Satellite Imagery
Roland Perko, Hannes Raggam, Karlheinz Gutjahr, Wolfgang Koppe, and Jürgen Janoth 87

Classification and Segmentation of Scanned Library Catalogue Cards using Convolutional Neural Networks
Matthias Wödlinger and Robert Sablatnig 90

Visual Odometry For Industrial Cable Laying
Ana Gregorac, Karlheinz Gutjahr, Richard Ladstädter, Roland Perko, and Wolfgang Höppl 92

Few-shot Object Detection Using Online Random Forests
Werner Bailer and Hannes Fassold 95

The Problem of Fragmented Occlusion in Object Detection
Julian Pegoraro and Roman Pflugfelder 98

A Centerline-Guided Approach for Aorta and Stent-Graft Segmentation
Bertram Sabrowsky-Hirsch, Stefan Thumfart, Richard Hofer, Wolfgang Fenz, Pierre Schmit, and Franz Fellner 102

Image Synthesis in SO(3) by Learning Equivariant Feature Spaces
Marco Peer, Stefan Thalhammer, and Markus Vincze 108

Frame Border Detection for Digitized Historical Footage
Daniel Helm, Bernhard Pointner, and Martin Kampel 114

Highly Accurate Binary Image Segmentation for Cars
Thomas Heitzinger and Martin Kampel 116

Powder Bed Analysis in Additive Manufacturing Using Image Processing
Florian Recla and Martin Welk 122

Grasping Point Prediction in Cluttered Environment using Automatically Labeled Data
Stefan Ainetter and Friedrich Fraundorfer 124

The Difficulties of Detecting Deformable Objects Using Deep Neural Networks
Nikola Djukic, Walter G. Kropatsch, and Markus Vincze 131

Border Propagation: A Novel Approach To Determine Slope Region Decompositions
Florian Bogner, Alexander Palmrich, and Walter G. Kropatsch 137

How High is the Tide? Estimation of Flood Level from Social Media
Julia Strebl, Djordje Slijepcevic, Armin Kirchknopf, Muntaha Sakeena, and Markus Seidl 143

Real-World Video Restoration using Noise2Noise
Martin Zach and Erich Kobler 145

Asymptotic Analysis of Bivariate Half-Space Median Filtering
Martin Welk 151

360° monitoring for robots using Time-of-Flight sensors
Thomas Maier and Birgit Hasenberger 157

Towards Identification of Incorrectly Segmented OCT Scans
Verena Renner and Jiří Hladůvka 159

Evaluating Counter Measures against SIFT Keypoint Forensics
Muhammad Salman and Andreas Uhl 166

Automated Generation of 3D Garments in Different Sizes from a Single Scan
Stefan Hauswiesner and Philipp Grasmug 172

Preface

The Joint Austrian Computer Vision and Robotics Workshop (OAGM and ARW Workshop) was originally planned to take place at Graz University of Technology on April 16 and 17, 2020. However, due to the COVID-19 situation in 2020, we had to do without a physical event and thus we had to cancel both the originally scheduled event in April 2020 and the re-scheduled event in September 2020. After the submission and reviewing process could be finished almost as planned, we decided to publish the proceedings anyway. In this way, many thanks to all who had made this possible (authors, reviewers, program chairs, publisher)!

The main intention of the joint workshop would have been to bring together researchers, students, professionals, and practitioners from the fields of Computer Vision and Robotics to present and actively discuss the latest research and developments. While in the past there has been a perceivable gap between these two research directions, we can recognize that there are more and more common interests, which can also be seen from contributions from both scientific communities. Overall, there have been 50 original contributions, where an international program committee selected 38 for publications based on a double-blind review process, where each paper was reviewed by three reviewers. To follow the tradition, outstanding contributions will be awarded prices sponsored by OCG (OAGM track) and IEEE RAS (ARW track). In addition, for the very first time, there will be an IEEE Women in Engineering Award for the best contribution of a female first author (both tracks). Moreover, we would like to thank Land Steiermark (Ressort für Wirtschaft, Tourismus, Europa, Wissenschaft und Forschung) and the Federal Ministry for Climate Action, Environment, Energy, Mobility, Innovation and Technology for supporing the Joint Austrian Computer Vision and Robotics Workshop 2020.

Even though there is no physical event, we would also like to thank the international speakers who accpected our invitations: Frank Kirchner (DFKI Bremen), Jürgen Gall (University of Bonn), and Thomas Schmickl (KFU Graz). We would be happy to re-invite them in future either for an upcoming physical or virtual event.

In this way, we are happy to make at least the proceedings available! Stay healthy, hopefully seeing you again in 2021,

Mathias Brandstötter, Gerald Steinbauer (ARW Chairs)

Friedrich Fraundorfer, Roland Perko, Peter M. Roth (OAGM Chairs)

Graz, July 2020

ARW Workshop Chairs

Gerald Steinbauer (Graz University of Technology)

Mathias Brandstötter (JOANNEUM RESEARCH)

OAGM Workshop Chairs

Friedrich Fraundorfer (Graz University of Technology)

Roland Perko (JOANNEUM RESEARCH, Graz University of Technology)

Peter M. Roth (Graz University of Technology)

Program Committee

Helmut Ahammer (Medical University of Graz)
Stefan Auer (German Aerospace Center)
Áron Ballagi (Szechenyi Istvan University)
Csaba Beleznai (Austrian Insitute of Technology)
Christian Bettstetter (Klagenfurt University)
Horst Bischof (Graz University of Technology)
Michael Bleyer (Microsoft)
Kristian Bredies (University of Graz)
Katja Bühler (VRVis)
Wilhelm Burger (Upper Austria University of Applied Sciences)
Bernhard Dieber (JOANNEUM RESEARCH)
Raimund Edlinger (University of Applied Sciences Upper Austria)
Christian Eitzinger (Profactor Research)
Cornelia Fermüller (University of Maryland)
Johannes Fürnkranz (Johannes Kepler University Linz)
Harald Ganster (JOANNEUM RESEARCH)
Margrit Gelautz (Vienna University of Technology)
Levente Hajder (Eötvös Loránd University)
Martin Humenberger (Naver Labs Europe)
Michael Janisch (BMLV)
Martin Kampel (Vienna University of Technology)
Martin Kandlhofer (Graz University of Technology)
Viktor Kaufmann (Graz University of Technology)
Walter G. Kropatsch (Vienna University of Technology)
Wilfried Kubinger (FH Technikum Wien)
Arjan Kuijper (TU Darmstadt)
Franz Kurz (German Aerospace Center)
Marco Körner (Technical University of Munich)
Wilfried Lepuschitz (Practical Robotics Institute Austria)
Mathias Lux (Klagenfurt University)
Martina Mara (Johannes Kepler University Linz)
Andreas Müller (JKU Johannes Kepler Universität Linz)
Clemens Mühlbacher (ARTI – Autonomous Robot Technology GmbH)
Gerhard Paar (JOANNEUM RESEARCH)
Justus Piater (University of Innsbruck)
Andreas Pichler (Profactor GmbH)
Horst Pichler (JOANNEUM RESEARCH)
Michael Reip (incubedIT GmbH)
Bernhard Rinner (Klagenfurt University)
Robert Sablatnig (Vienna University of Technology)
Josef Scharinger (Johannes Kepler University)
Thomas Schmickl (University of Graz)
Lukas Silberbauer (taurob OG)
Darko Stern (Medical University of Graz)
Gernot Stübbl (Profactor GmbH)

Stefan Thumfart (RISC Software GmbH)
Andreas Uhl (University of Salzburg)
Martin Welk (Private University of Health Sciences, Medical Informatics and Technology)
Christopher Zach (Chalmers University of Technology)
Michael Zillich (Blue Danube Robotics)

OAGM Awards 2019

The

OCG Best Student Paper Awards 2019

were awarded to the papers

On the Use of Artificially Degraded Manuscripts for Quality Assessment of Readability Enhancement Methods

by

Simon Brenner and Robert Sablatnig.

and

Efficient Multi-Task Learning of Semantic Segmentation and Disparity Estimation

by

Robert Harb and Patrick Knöbelreiter.

The

OCG Best Poster Award 2019

was awarded to the paper

Combining Deep Learning and Variational Level Sets for Segmentation of Buildings

by

Muntaha Sakeena and Matthias Zeppelzauer.

ARW Awards 2019

The

IEEE RAS Austria Best Research Paper Award

was awarded to the paper

A Dynamical System for Governing Continuous, Sequential and Reactive Behaviors

by

Raphael Deimel.

The

Best Poster Award

sponsored by the ABB-Group was awarded to the paper

Multilingual Speech Control for ROS-driven Robots

by

Dominik Hofer, Simon Brunauer and Hannes Waclawek.

Index of authors

Aburaia, Mohamed, 2, 13
Ainetter, Stefan, 50, 124
Arneitz, Fred, 50
Arronde, Dailys, 50

Bailer, Werner, 95
Ballagi, Áron, 48
Baumann, Wilfried, 19
Bieber, Ronald, 19
Bogner, Florian, 137
Brandstötter, Mathias, 53

Debiasi, Luca, 65
De Bortoli, Marco, 34
Dhakate, Rohit, 50
Djukic, Nikola, 131

Ehardt-Schmiederer, Margit, 19

Fabricius, Felicitas, 34
Fassold, Hannes, 95
Fellner, Franz, 102
Fenz, Wolfgang, 102
Fraundorfer, Friedrich, 50, 124

Gattringer, Hubert, 59
Gebser, Martin, 34
Gietler, Harald, 50
Grabner, Alexander, 71
Grasmug, Philipp, 172
Gregorac, Ana, 92
Gubensäk, Wolfgang, 50
Gutjahr, Karlheinz, 87, 92

Hajdu, Csaba, 48
Hasenberger, Birgit, 157
Hauswiesner, Stefan, 172
Heitzinger, Thomas, 116
Helm, Daniel, 114
Hettegger, Daniel, 31
Hirschmanner, Matthias, 42
Hladůvka, Jiří, 159
Hoebert, Timon, 27
Hofbauer, Heinz, 65
Hofer, Richard, 102
Hollenstein, Jakob, 29
Höppl, Wolfgang, 92

Jamadi, Ali, 42
Janoth, Jürgen, 87

Kampel, Martin, 114, 116
Kandlhofer, Martin, 19
Kauba, Christof, 65
Kirchknopf, Armin, 143
Kniewasser, Günther, 71
Knöbelreiter, Patrick, 79
Kobler, Erich, 145
Komenda, Titanilla, 53
Koppe, Wolfgang, 87
Kriegler, Andreas, 6
Kropatsch, Walter G., 131, 137
Krög, Gabriel, 59
Kubinger, Wilfried, 4, 21

Ladstädter, Richard, 92
Lepuschitz, Wilfried, 27
Lichtenecker, Daniel, 59

Maier, Thomas, 157
Maximilian, Selmair, 34
Medeiros, Mylena, 50
Menzinger, Manuel, 19
Merdan, Munir, 27
Müller, Andreas, 59

Neuberger, Bernhard, 42
Novotny, Georg, 21

Palmrich, Alexander, 137
Papa, Maximilian, 4
Patten, Timothy, 42
Peer, Marco, 108
Pegoraro, Julian, 98

Perko, Roland, 87, 92
Permann, Alexander, 31
Pflugfelder, Roman, 98
Piater, Justus, 29
Pointner, Bernhard, 114
Prommegger, Bernhard, 65

Raggam, Hannes, 87
Rathmair, Michael, 53
Rauer, Johannes, 2
Rebol, Manuel, 79
Recla, Florian, 122
Reip, Michael, 34
Renaudo, Erwan, 29
Renner, Verena, 159
Roth, Peter M., 71

Sablatnig, Robert, 90
Sabrowsky-Hirsch, Bertram, 102
Sakeena, Muntaha, 143
Salman, Muhammad, 166
Sattinger, Vinzenz, 4
Saveriano, Matteo, 29
Schmit, Pierre, 102
Seidl, Markus, 143
Slijepcevic, Djordje, 143
Steigl, Dominik, 13
Steinbauer, Gerald, 19, 31, 34
Steiner, Martin, 53
Stetco, Christian, 50
Strebl, Julia, 143

Thalhammer, Stefan, 108
Thumfart, Stefan, 102

Uhl, Andreas, 65, 166

Vincze, Markus, 42, 108, 131

Weiss, Stephan, 50
Welk, Martin, 122, 151
Winkler, Thomas, 19
Wöber, Wilfried, 2, 6, 13
Wödlinger, Matthias, 90

Zach, Martin, 145
Zangl, Hubert, 50

Austrian Robotics Workshop

Semi-Automatic Generation of Training Data for Neural Networks for 6D Pose Estimation and Robotic Grasping

Johannes Nikolaus Rauer, Mohamed Aburaia, Wilfried Wöber
FH Technikum Wien
{rauer,aburaia,woeber}@technikum-wien.at

Abstract. *Machine-learning-based approaches for pose estimation are trained using annotated ground-truth data – images showing the object and information of its pose. In this work an approach to semi-automatically generate 6D pose-annotated data, using a movable marker and an articulated robot, is presented. A neural network for pose estimation is trained using datasets varying in size and type. The evaluation shows that small datasets recorded in the target domain and supplemented with augmented images lead to more robust results than larger synthetic datasets. The results demonstrate that a mobile manipulator using the proposed pose-estimation system could be deployed in real-life logistics applications to increase the level of automation.*

1. Introduction

Production facilities have successfully deployed classic fixed-programmed robots since the 1960s. Due to their inability to perceive the environment, such robots have mostly been used in mass production, where a static setup can be assumed [8]. The production industries' move away from mass production towards highly customized goods requires increased flexibility. Deploying mobile manipulators, a combination of mobile and articulated robots, for intra-logistical transport tasks, promises this desired modularity [6]. Since the accuracy achieved by mobile robot navigation is not sufficient to grasp objects, robots need sensors to perceive their surroundings and autonomously detect objects' poses [1]. The most promising approaches for pose estimation are machine-learning-based methods applied to camera data [2]. Deep neural networks are trained using annotated ground-truth data – images showing the object and information of its pose [4]. State-of-the-art methods for creating such data use markers rigidly attached to the objects, which have to be removed in cumbersome post-processing [3], or need human annotators that align 3D models to video-streams [5]. In this work an approach to semi-automatically generate 6D-pose-annotated training data using an articulated robot is presented.

2. Semi-Automatic Data-Generation

As shown in Figure 1 the object is placed in front of the robot and a fiducial marker is put on it in a defined pose. The pose of the marker with respect to the camera is computed from the captured image and used to calculate the pose of the object with respect to the robot's base. The marker is captured from multiple perspectives and the mean pose is calculated to minimize errors of the camera calibration and marker detection. Afterwards the marker is removed (care must be taken that the object is not displaced) and the robot arm moves around the object to capture images and associated object-pose data automatically. In order to make the data also usable for training neural networks for object detection, the object can be rendered in a virtual environment to calculate segmentation masks. The design minimizes the extent of human labor. It is only necessary to place the marker on the object, capture images of it, and remove it again, to enable recording of several thousand training images fully autonomously. Drawbacks are that the process has to be repeated to cover the other half of the orientation space and that the background is static. However, this can be solved by data augmentation.

3. Results & Discussion

Multiple annotated datasets are created using the proposed method and used to train the deep-learning-based 6D pose estimation system DOPE [7]. The annotated training data is split into five equally sized

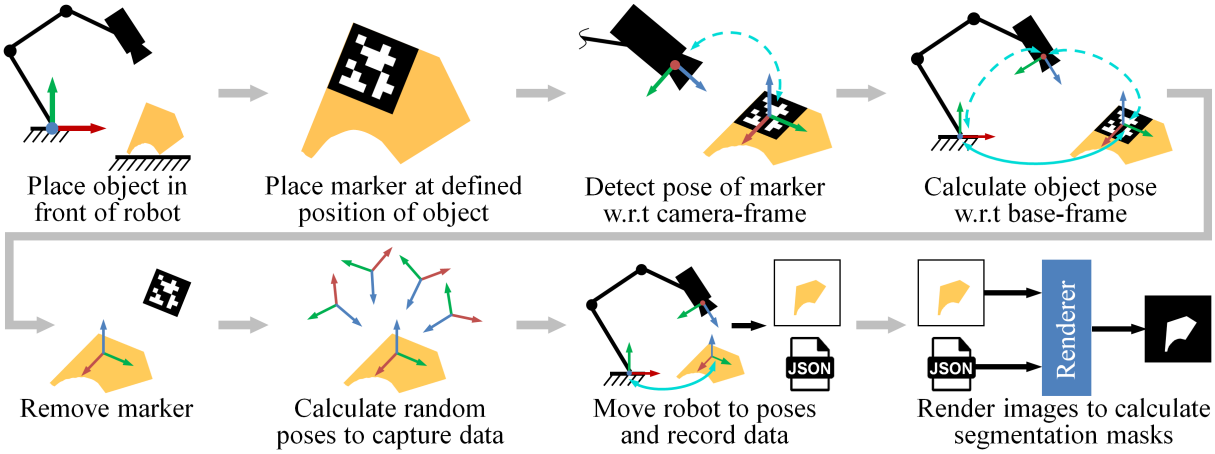


Figure 1. Procedure for generating annotated data, using a robot and a movable fiducial marker.

portions and merged to gain datasets containing 20% to 100% (15k images) of all recorded samples.

The translational-15mm-error metrics (percentage of tested data for which the translational error is smaller than 15 mm – accuracy necessary for grasping) [7] in Figure 2 show, that using pre-trained models (blue, 6-10) leads to better performance than initializing networks with random weights (red, 1-5). Bigger datasets do not necessarily improve the accuracy since biased datasets lead to wrong generalizations (e.g. network 5). A relatively small dataset recorded in the target domain achieves better results than a several times larger synthetic dataset (network 12: 15k real + 15k domain randomized images), especially when extended using data augmentation (network 11: smallest real dataset augmented twice). The rotational errors show similar results, but are generally lower.

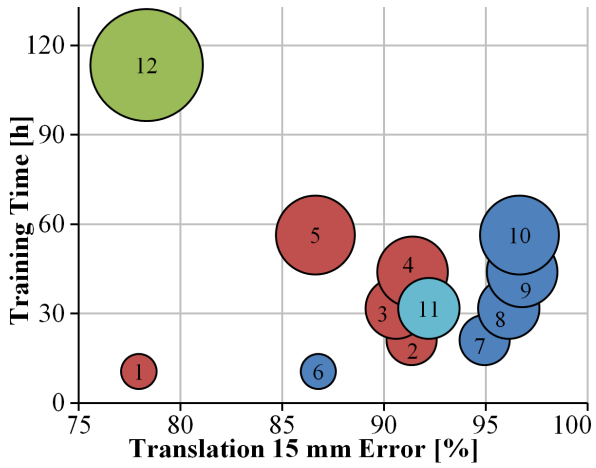


Figure 2. Translational errors compared regarding training time: Synthetic data (green), augmented data (cyan), pre-trained (blue) and non-pre-trained networks (red). Bubble-size visualizes dataset-size.

A qualitative evaluation using a real mobile manipulator confirms that the proposed pose-estimation system could be deployed in real-life logistics applications to increase the level of automation.

References

[1] U. Asif, M. Bennamoun, and F. A. Sohel. RGB-D object recognition and grasp detection using hierarchical cascaded forests. *IEEE Transactions on Robotics*, 33(3):547–564, 2017.

[2] G. Du, K. Wang, and S. Lian. Vision-based robotic grasping from object localization, pose estimation, grasp detection to motion planning: A review. *CoRR*, 2019.

[3] M. Garon, D. Laurendeau, and J. F. Lalonde. A framework for evaluating 6-DOF object trackers. In *15th European Conference on Computer Vision – ECCV*, pages 608–623, 2018.

[4] I. J. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, Cambridge, MA, USA, 2016.

[5] P. Marion, P. R. Florence, L. Manuelli, and R. Tedrake. Label Fusion: A pipeline for generating ground truth labels for real RGBD data of cluttered scenes. In *IEEE International Conference on Robotics and Automation – ICRA*, pages 1–8, 2017.

[6] D. Pavlichenko, G. M. García, S. Koo, and S. Behnke. Kittingbot: A mobile manipulation robot for collaborative kitting in automotive logistics. In *15th International Conference on Intelligent Autonomous Systems – IAS*, pages 849–864, 2018.

[7] J. Tremblay, T. To, B. Sundaralingam, Y. Xiang, D. Fox, and S. Birchfield. Deep object pose estimation for semantic robotic grasping of household objects. In *2nd Annual Conference on Robot Learning – CoRL*, pages 306–316, 2018.

[8] J. Wallén. The history of the industrial robot. *Technical report from Automatic Control at Linköpings universitet*, 2008.

Feasibility study of a certifiable production environment using safe environmental sensor systems

Maximilian Papa, Vinzenz Sattinger, Wilfried Kubinger
UAS Technikum Wien, Hoeststaedtplatz 6, A-1200 Vienna
{maximilian.papa,vinzenz.sattinger,wilfried.kubinger}@technikum-wien.at

Abstract. *Safe robot development is based on three factors: safety, performance and economy. Currently however, only two properties can be maximized at once, which is why an alternative for maximizing all three factors has been worked on. In particular the topic of safe environment has been discussed, where sensors from individual robots will be relocated in the environment. A sensor thus monitors more than one robot, which leads to an increase in efficiency. Due to the novelty, technical and legal requirements of such a system have first been clarified. The required components have then been determined in order to plan a possible implementation. Finally an adapted concept for the Digital Factory of the UAS Technikum Vienna showed the feasibility of a safety-certifiable environmental sensor system.*

1. Introduction

The fourth industrial revolution is characterized by a flexible production of individual products, which will be ensured by interconnected smart components and robots [5]. However, robots of the third industrial revolution are not flexible enough for this task due to their stationary location. For this reason they are accompanied by mobile robots, which enable a higher production flexibility [3]. These robots will further relieve human work forces from monotonous work allowing them to work on more complex tasks [2]. A combination of human flexibility and robot repeatability thus represents the future of production, whereby safe cooperation must be guaranteed. Furthermore, appropriate changes in intelligent production systems are recommended due to an expected compound annual growth of 23.1% between 2018 and 2023 in the area covering technologies of the fourth industrial revolution [4].

2. Current Solutions and Motivation

Development of industrial and mobile robots is based on three factors as shown in Figure 1. However, current concepts only manage to maximize two of the three properties at once. As safety should never be neglected in a factory, there are two possible configurations: Either expensive systems or cost-efficient approaches with weaker performance [1].

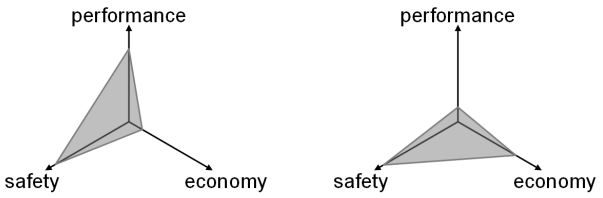


Figure 1. Factors of common robot development

The motivation of this work is the maximization of all three properties in development of safe robotics. A promising solution would be the relocation of individual robot sensors into the environment. The basic idea is that one sensor can monitor several robots and therefore fewer sensors are needed.

3. Basic Requirements and Methods

As there is no safe environment system available yet, technical and legal requirements have to be researched for safety certification in Austrian enterprises first. However, applicable documents differ in each case and experts (e.g. TÜV AUSTRIA, labour inspectorate, etc.) should be consulted for support. Afterwards suitable components of the three identified key elements of sensor systems, processing units and communication modules have to be researched and compared. Subsequently concepts with these requirements and components have to be created and a value benefit analysis should determine the best one.

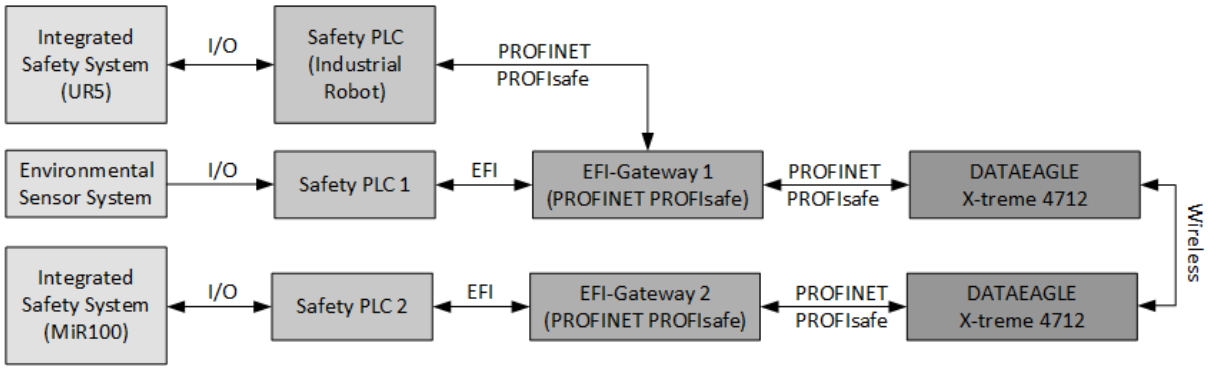


Figure 2. Connections between all components for the safe environment

4. Concepts of a safe environment

A common safety concept (Concept 1) using separate sensors in each robot represents the state-of-the-art [1]. The total close-down of the factory (Concept 2) is another option, but entrances must be monitored to guarantee that no human can enter the working factory. People may only enter after switching the factory to a collaborative mode or a safety stop. If there were people working in the factory regularly, a division of the factory into different segments (Concept 3) would be better. Only segments in which persons are located have to work in collaborative mode, making this concept more efficient. However, this concept is not optimal with many people working in it either. Consequently monitoring the whole environment with active (Concept 4) or passive (Concept 5) person detection is recommended. Workers have to wear a transmitter on their body for the active variant which is not required in the passive detection.

5. Results and Discussion

Safety-certified components are already existing for the first three concepts, but not for active/passive detection. However, the first concept by purchasing safety-certified components for each robot would also be the most expensive concept. Active/passive person detection also requires various components to be installed throughout the factory. Only the second and third concepts would require few sensors at the zone entrances, making them very economical but also inefficient with a high volume of people.
Based on these facts and the emphasis on safety, economy and performance, the third concept was chosen for an implementation in the Digital Factory. Of course, the optimal choice depends above all on the size of the environment and the number of robots.

6. Summary and Outlook

Based on these results a detailed safe environment implementation for the Digital Factory has been planned, where the safe communication (shown in Figure 2) represents the centerpiece of the system: Environmental sensors have thereby been connected to inputs/outputs of a safety PLC, which is further connected to the other safety PLCs via PROFIsafe by PROFINET. However, already existing PLCs do not had a PROFIsafe interface, which is why EFI-gateways have been included. Furthermore special DATAEAGLE modules are required for a safe wireless connection to mobile robots.

A realization of the safe environment is therefore actually possible, but the safe environment concept will probably only become a serious alternative with the development of safety-certified components for active/passive human detection.

References

[1] M. Arndt. *Safe and Cost-Efficient Mobile Robot Navigation in Aware Environments*. PhD thesis, Technische Universität Kaiserslautern, 2016.

[2] Automations Praxis. Mobile Robotik löst langwierigen Transport, 2017. [Online]. Available: https://automationspraxis.industrie.de/servicerobotik/mobile-robotik-loest-langwierigen-transport/ [Accessed: 29.12.2018].

[3] R. Siegwart and I. R. Nourbakhsh. *Introduction to Autonomous Mobile Robots*. MIT Press, USA, 2004.

[4] M. Sullivan. Industry 4.0 Technologies: Global Market Through 2023, 2018. [Online]. Available: https://www.bccresearch.com/market-research/manufacturing/ [Accessed: 16.06.2019].

[5] B. Vogel-Heuser, T. Bauernhansl, and M. Ten Hompel. *Handbuch Industrie 4.0 Band 4*, volume 2. Springer Berlin Heidelberg, Berlin, Heidelberg, 2017.

Vision-based Docking of a Mobile Robot

Andreas Kriegler, Wilfried Wöber
UAS Technikum Vienna
{mr18m016,woeber}@technikum-wien.at

Abstract. *For mobile robots to be considered autonomous they must reach target locations in required pose, a procedure referred to as docking. Popular current solutions use LiDARs combined with sizeable docking stations but these systems struggle by incorrectly detecting dynamic obstacles. This paper instead proposes a vision-based framework for docking a mobile robot. Faster R-CNN is used for detecting arbitrary visual markers. The pose of the robot is estimated using the solvePnP algorithm relating 2D-3D point pairs. Following exhaustive experiments, it is shown that solvePnP gives systematically inaccurate pose estimates in the x-axis pointing to the side. Pose estimates are off by ten to fifty centimeters and could therefore not be used for docking the robot. Insights are provided to circumvent similar problems in future applications.*



Figure 1. The visual target used for docking. The target location is on the ground infront. The origin for the PnP solvers is in the upper left corner. The logos are roughly 9x3 centimeters in size. The upper right logo was raised during experiments to remove coplanarity.

1. INTRODUCTION

Docking can be understood as the localization and navigation of a robot towards a target location [1]. In contrast to path-planning across larger distances, docking does not require obstacle avoidance methods but instead seeks highly accurate pose estimates [28]. As long as the pose of the robot and the target location are known in a reference coordinate system path planning algorithms can easily generate control commands. In the xy ground-plane, the pose $\vec{x}$ consists of three degrees of freedom, x, y, and θ as the rotation about its own axis z, and is described using the state at time t

$$\vec{x}_t = \begin{pmatrix} x & \dot{x} & y & \dot{y} & \theta & \dot{\theta} \end{pmatrix}_t^T \tag{1}$$

where $\dot{x}$, $\dot{y}$ and $\dot{\theta}$ describe the speed of the robot in x and y and its rotation respectively. As Thrun *et al*. [32] write outlining the motion model and measurement model, taking multiple control steps $\vec{u}_t$ with only an initial measurement or observation $\vec{z}_t$ leads to large uncertainties about its pose, they propose a measurement step after every control to restore confidence in the belief $bel(\vec{x})$. These measurements can be non-vision methods such as evaluating detections from LiDAR-scans [22] or can come from a camera setup providing visual feedback [6]. Yurtsever *et al*. [34] show in their survey on automated driving systems (ADS) that computer vision (CV) based approaches to navigation have become increasingly popular. Artificial landmark detection as described by Luo *et al*. [19] and gradient based optical flow [20] rival modern non-vision solutions. Classical non-vision systems typically employ LiDAR technology, indoor GPS or wireless fingerprinting [17]. While LiDARs are still widely used commercially (such as MiRs and Robotinos) recent advances in deep learning and their application in the ADS domain are of more scientific interest. Deep Convolutional Neural Networks (CNNs) have proven successful at tackling a variety of perception problems, including object detection [26] and pose esti-

mation [30]. Open source implementations for different learning tasks are plentiful and can be used to provide perception for a robotics system. Due to the strong capabilities of CNNs as general feature extractors, it is possible to learn multiple visual targets which can be different depending on the environment or application. This relaxes the constraint of using specifically designed visual markers that classical CV methods pose. The learning task of the object detector in this work is comparatively simple (only one class of logos exist and they are easily distinguishable from the rest of the target, see Fig.1).
In previous work the LiDAR of the mobile robot, a robotino, was used to create a map of the environment and localization was implemented with the amcl package. While this pipeline in combination with obstacle avoidance methods has been useful for path-planning across the room, only employing the AMCL the robot arrives at the target position with great inaccuracy (10cm to 20cm). Therefore, for this project an entirely vision-based solution for docking was developed which is bound to take over the task of generating pose-estimates from the AMCL once the robot comes close to the docking target.
The aim of this work therefore is to approach and dock onto desired targets in a semi industrial environment with sufficiently high accuracy. To contribute to the transition of state-of-the-art CNNs from public datasets to real world problems an appropriate combination of old and new algorithms is presented in this work. A CNN based object detectors is used for image processing and object detection, followed by a camera pose estimation algorithm using point correspondences from the detections. The presented method could be easily adapted to learn new target positions outfitted with a visual marker with minimal setup requirements.

2. STATE OF THE ART

The problem of estimating the pose of a calibrated camera, assuming a known 3D scene, is known as the PnP-problem [29]. The idea is to use a feature detector such as SIFT [16] or SURF [2] to extract features from multiple sequential images. Since an image of a known 3D point gives two nonlinear constraints on camera pose and calibration, using three points (or more precisely three image-object point pairs) would give all 6 pose parameters. As [33] point out, such minimal cases lead to polynomial systems with multiple solutions, hence one additional point is used. This leads to four necessary points for estimating the pose (and one intrinsic parameter) and six points for estimation of 3D pose and five additional calibration parameters. The problem is formulated diffently for the planar two-dimensional or the general, aforementioned three dimensional case. Direct Linear Transformation (DLT, [9]) allows the estimation of the homography matrix $\boldsymbol{H}$ for the planar problem, requiring at least four 2D-3D point correspondences. For the general case, DLT estimates the projection matrix $\boldsymbol{P}$ and requires at least six such correspondences. In either case, $\boldsymbol{H}$ or $\boldsymbol{P}$ can be expressed with a set $\boldsymbol{A}\vec{x} = 0$ of multiple pairs of independent equations. Since individual pixels are generally noisy, no exact solution can be obtained using DLT, only an approximate solution by obtaining the SVD of $\boldsymbol{A}$. It should be noted, that for the noisy and overconstrained case, only the eigenvector of $\boldsymbol{A}^T\boldsymbol{A}$, corresponding to the smallest eigenvalue, should be computed. A continuation to DLT is the family of PnP algorithms. Efficient PnP or EPnP [14] uses the notion that each of the n 3D-2D point pairs are expressed as weighted sum of four virtual control points, and solves the pose problem from these control points. Perspective-Three-Point or P3P is a method applicable if only three correspondences are obtained, and in turn returns four real, possible solutions, the newest implementation being Lambda Twist P3P [25]. A fourth point pair can be used to remove this four-solution ambiguity.
Kartoun *et al*. [12] were able to achieve docking times averaging 85 seconds but attributed the success of their method to the unique hardware on the robot and a generously large docking station. Burschka *et al*. [3] take the aforementioned approach to the outdoors, using a Kanade-Lucas tracker [18] to track points in image sequences, followed by RANSAC and DLT. They achieve good results for rotation, but struggle with estimating translation. In the work of Mehralian *et al*. [21] an Extended Kalman Filter (EKF, [11]) is combined with PnP algorithms to create EKFPnP. They achieve better robustness against noisy features, although no details are given regarding the feature tracker.

In the field of deep learning, pose estimation is a well researched problem [23], camera pose estimation is less so [13] and no architectures or datasets exists specifically designed for docking a mobile robot. The dataset would need to include the complete pose of the robot for every captured image to allow end-

to-end training. Instead, Shalnov *et al*. [30] were able to create a deep model using a CNN for camera pose estimation via object detections of human heads. In the work of Pavlakos *et al*. [24] a geometric approach to object pose estimation using semantic keypoints is taken but their published dataset only uses outdoor objects and is thus not applicable to docking. Lastly, as part of Zhou *et al*. [35]'s Centernet, they are proposing to regress from centerpoints to other object properties including pose but their framework is unnecessarily complex for the task at hand.

While the methods are numerous, no single framework exists that combines deep learning for object detections with a PnP-solver, all for the application of mobile robot docking. This work shows the hesitation of using CNNs for robot docking is unwarranted, as long as the learning task is managable in complexity.

3. METHODS AND IMPLEMENTATION

The robotino mobile robot used in this project was equipped with a Logitech C920 USB webcam. A remote desktop with an NVIDIA GTX 1080 GPU runs ROS to control the robot and process the images.

To showcase the flexibility of the pipeline regarding the visual target, no QR-tags or ARUCO markers [8, 27] were used. Three small paper printouts of a logo were instead fixed on a board roughly 20 by 15 centimeters in size and this board was used for training the detector. Video data was collected while arbitrarily moving the robot around close to the target. From the roughly 4500 recorded images 100 were selected to form the training set. The chosen images show the target from different viewing angles, distances, lighting conditions while a few images do not show the target at all to control for false positives. The bounding box coordinates of the three logos in all 100 images were manually annotated. Creating annotations took around three hours to complete. Resizing the images to 512x512 RGB-images allows the usage of Che *et al*. [4]'s toolbox with many different object detectors implemented.

Accuracy of the detector is important, since wrong detections would lead to wrong pose estimates and erroneous controls, while inference speed is important to enable a smooth docking, although inference times below 70 milliseconds are unnecessary, due to the bottleneck imposed by transporting 960x720 images from the camera to the remote desktop via Wi-Fi using the ROS image_transport package for compressed transfer. Looking at various speed-accuracy tradeoff comparisons between object detectors, Faster R-CNN [26] with pretrained ResNet [10] backbones seems to be a sweet spot, ResNet50 was chosen for this implementation. Faster R-CNN belongs to the class of detectors using a separate region proposal network to generate bounding box proposals. For the optimizer the default stochastic gradient descent with momentum of 0.9 was used and learning rate was kept default at 0.01 with a linear step learning rate scheduler and warmup. Other parameters and image augementation steps were kept default to Che *et al*. [4]'s configuration of Faster R-CNN for PascalVOC [5], including a 50 percent chance of a random horizontal flip. From the infered bounding boxes, image coordinates of the upper-left and lower-right corners of all three logos are saved for the PnP-solver. The Faster R-CNN network was trained for fifty epochs which amounted to 37 minutes training time on a GTX 1080 graphics card. GPU memory usage was 2GB showing a weaker graphics unit would suffice. Both bounding box and classification loss plateued after training for ten epochs.

The required pose estimate at timestep t for path planning can be described with the transformation matrix $\mathbf{T}_{base,t}^{target} \in \mathbb{R}^{4x4}$ from the base link of the robot to the target position near the station

$$\mathbf{T}_{base,t}^{target} = \begin{bmatrix} \mathbf{R} & \vec{t} \\ \vec{0} & 1 \end{bmatrix}_t \tag{2}$$

with $\mathbf{R} \in \mathbb{R}^{3x3}$ and $\vec{t} \in \mathbb{R}^{3x1}$ being the rotation matrix and translation vector to be estimated at sampling time t respectively. Physically measuring the transformation from the base link of the robot to the camera sensor as well as relating the logos at $\mathbf{K}_{logo}$ to the target location allows an estimated camera pose from a reference coordinate system on the logo-board $\mathbf{T}_{logo}^{camera}$ to be linearly tranformed into $\mathbf{T}_{base}^{target}$. Getting the transformation $\mathbf{T}_{logo}^{camera}$ with a calibrated camera and assuming the pinhole camera model means solving correspondences of points in 2D image space and those same points in the 3D real world. After calibrating the camera using the ROS *camera_calibration* package, the measured points in the object frame and saved image coordinates are combined in the Open-CV *solvePnP* algorithm using the intrinsic camera parameters. Available variations of the algorithm are *iterative*, which is the default method based on Levenberg-Marquardt optimization [15] to find a pose which minimizes reprojection er-

ror (sum of squared distances), *P3P* based on [7] which requires only four of the six point pairs and *EPnP* mentioned earlier. All three variations were tried and tested. The estimated rotation and translation vectors, after using *Rodrigues* to transform the rotation vector into the rotation matrix $\mathbf{R}$, form $\mathbf{T}_{logo}^{camera}$ and therefore finally $\mathbf{T}_{base,t}^{target}$. As Siegwart *et al.* [31, p. 81ff] write, desired velocity can then easily be generated using estimated parameters k_ρ and k_α for a linear controller.

The entire pipeline can be quickly summarized as follows:

1. Create the visual target with arbitrary logos. Physically measure the logo corners and their position in relation to T_{logo}. Relate T_{logo} to T_{target} and on the robot T_{camera} to T_{base}.

2. To avoid bias in data collection, implement a random-walk in logo vicinity but constrain θ to enable the camera to face the logo most of the time. Annotate bounding-box coordinates of the logos for select images.

3. Train the Faster-RCNN object detector with this dataset. For docking, load the model and obtain bounding-boxes using ROS image-callbacks.

4. Use the inferred coordinates together with the measurements and intrinsic parameters of the camera in SolvePnP to obtain $T_{base,t}^{target}$ at every timestep t.

5. Use a simple linear controller to generate ROS motion control commands to guide the robot towards the docking target.

4. RESULTS AND DISCUSSION

During inference, processing a single image within the ROS pipeline takes the detector approximately 35ms. On average the detection would result in five bounding box proposals, sorting by confidence and extracting the top three boxes gives six image coordinates close to the ground thruth typically within one to four pixels. Evaluating the mIoU gives 96.3% for thirteen test images. Object detection results are therefore both accurate and confident. The PnP-solver, the second major component of the framework, proved to be more troublesome producing inaccurate results. All three implementations of the solvePnP algorithm express the translation vector $\vec{t}_{logo}^{camera}$ using the right-hand coordinate system $\mathbf{K}_{logo}$. Preliminary results quickly showed that all methods are accurate in estimating y and z translation, but struggle with the x coordinate. To get a better understanding of the pose estimates, in particularly the estimated translation vector, an extensive field study was conducted. The robot was steered towards six points and the ground truth translation and rotation were noted. These poses are described by $\mathbf{K}_{idx}$ in Fig 2 where $idx \in \{dock, amcl, left_close, left_far, right_close, right_far\}$. At each point fifteen images were captured, supplied to the Faster-RCNN model and the obtained image coordinates from bounding boxes, specifically six points per image, given to the PnP-solvers. After first results were analyzed, showing again large errors in x, changes were made in hopes of achieving more accurate pose results. In particular, the following major changes were made:

1. The upper right logo was raised from the plastic board to remove the coplanarity of all six points. By removing the coplanarity more information is available for estimating the camera pose [9].

2. Since solvePnP, unlike regular DLT, does not estimate intrinsics, they are a possible cause of error. The camera was recalibrated and the new parameters used. The focal lengths and distortion coefficients differed slightly.

3. The autofocus of the camera was turned off. Captured images were still sharp and logos clearly visible nonetheless.

Afterwards, the same study was undertaken, capturing sequences of fifteen images at six locations, and using the detector followed by solvePnP to obtain camera pose estimates again. The translation vectors were than saved and subsequently plotted to give a visual representation of the results. Figure 2 shows the results of this experiment in a 3D plot. The most notable thing here is the *iterative* algorithm flipped the signs for all three axis in almost all estimates. Its results are therefore point-symmetrical about the origin, a known issue when using solvePnP. Also of note is that the large error in the x direction still persists. This error occurs throughout all experiments and is not intuitive; the estimates in x are strangely placed. All points lie on the negative (right) half plane (with exception of some *iterative* estimates), but the estimates for the locations with ground-truth in the

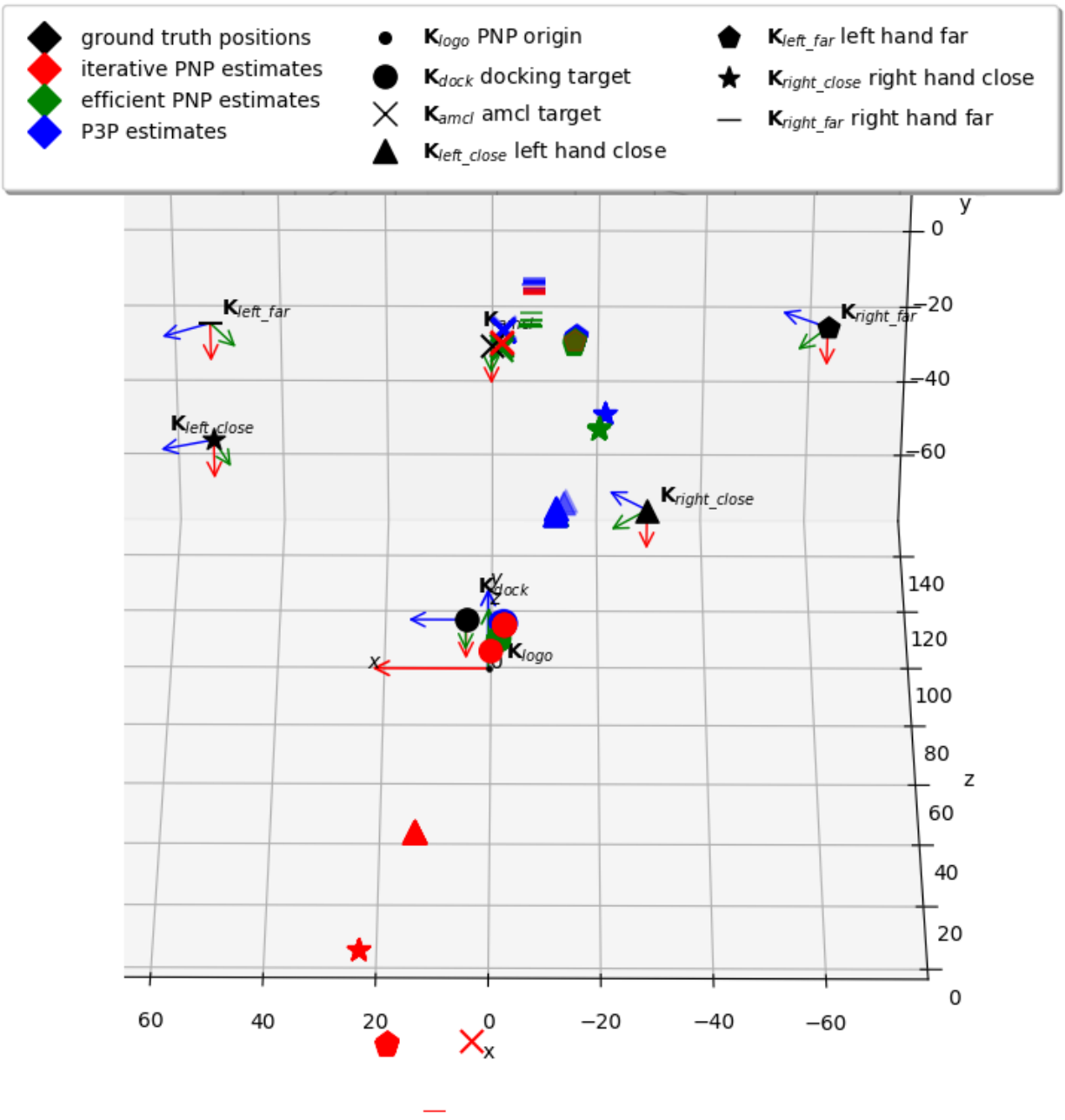


Figure 2. Pose estimates using the OpenCV solvePnP algorithm, for images captured at described six locations. Different colours are for different methods with black being ground truth locations. Different symbols mark the six different locations. Units are in centimeters. The 6 coordinate systems give the pose of the robot. The accuracy in estimating y and errors in x are similar as with the previous experiment. The point symmetry about the origin for the iterative algorithm is visible, having incorrectly flipped all three axis signs. Variance stays largely the same even with larger z.

left half are not simply mirrored across the z-axis. The distance gets consistently underestimated yet it seems with larger absolute value of x in ground-truth the absolute estimates in x also seem to increase. The estimates for the location $\mathbf{K}_{left_far}$ break this pattern, being very close to the estimates for $\mathbf{K}_{amcl}$, the location where the vision based navigation is supposed to take over after using amcl localization. It can also be observed that variance only slightly increases about the estimates in z with increasing z distance. Clusters are very compact, an improvement compared to the first experiment. This can be attributed to using more precise intrinsic camera parameters. It is also visible that all algorithms are accurate for estimating the small offset in y.

Unfortunately, the reason for this seemingly sys-

tematic error in x could not be determined as of yet but considering the flipped signs for almost all estimates made by the *iterative* method, numeric instability is likely to contribute to the fragile nature of the solvePnP class.

5. SUMMARY AND OUTLOOK

In this work a novel framework for docking a mobile robot using only vision-based sensors and algorithms was developed. A CNN based object detector yielded bounding boxes of logos with high accuracy and confidence. Measurements of the logos were taken and related in a coordinate system. The family of solvePnP algorithms implemented in OpenCV was used to estimate the camera pose using the detector results and intrinsic parameters. All methods consistently estimated wrong distances in one of the directions, namely the x-axis. Following preliminary experiments, changes were made, in particular the coplanarity of the object points was removed and recalibration of the camera undertaken, and the same experiments run again. Unfortunately the errors persisted, although improvements regarding the scatterness of the pose estimates could be made. Consequently, no control commands were generated and docking of the robot could not take place in this instance. For future reference, it is important to note the fragility of the solvePnP algorithms. The source of the errors is unclear and while additional point pairs could improve results regarding compactness, it seems unlikely they could alleviate the large errors in predicting the x coordinates.

References

[1] F. Alijani. Autonomous vision-based docking of a mobile robot with four omnidirectional wheels, 01 2017. Master's thesis.

[2] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In *European conference on computer vision*. Springer, 2006.

[3] D. Burschka and E. Mair. Direct pose estimation with a monocular camera. In *International Workshop on Robot Vision*. Springer, 2008.

[4] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu, Z. Zhang, D. Cheng, C. Zhu, T. Cheng, Q. Zhao, B. Li, X. Lu, R. Zhu, Y. Wu, J. Dai, J. Wang, J. Shi, W. Ouyang, C. C. Loy, and D. Lin. MMDetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019.

[5] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2), 2010.

[6] M. Fichtner and A. Grobmann. A probabilistic visual sensor model for mobile robot localisation in structured environments. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, volume 2, pages 1890–1895. IEEE, 2004.

[7] X.-S. Gao, X.-R. Hou, J. Tang, and H.-F. Cheng. Complete solution classification for the perspective-three-point problem. *IEEE transactions on pattern analysis and machine intelligence*, 25(8), 2003.

[8] S. Garrido-Jurado, R. Munoz-Salinas, F. J. Madrid-Cuevas, and R. Medina-Carnicer. Generation of fiducial marker dictionaries using mixed integer linear programming. *Pattern Recognition*, 51, 2016.

[9] R. Hartley and A. Zisserman. Multiple view geometry in computer vision second edition. *Cambridge University Press*, 2000.

[10] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. IEEE, 2016.

[11] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Fluids Engineering*, 82(1), 03 1960.

[12] U. Kartoun, H. Stern, Y. Edan, C. Feied, J. Handler, M. Smith, and M. Gillam. Vision-based autonomous robot self-docking and recharging. In *2006 World Automation Congress*. IEEE, 2006.

[13] A. Kendall and R. Cipolla. Geometric loss functions for camera pose regression with deep learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2017.

[14] V. Lepetit, F. Moreno-Noguer, and P. Fua. Epnp: An accurate o (n) solution to the pnp problem. *International journal of computer vision*, 81(2), 2009.

[15] K. Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of applied mathematics*, 2(2), 1944.

[16] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2), 2004.

[17] J. Y. Lu and X. Li. Robot indoor location modeling and simulation based on kalman filtering. *EURASIP Journal on Wireless Communications and Networking*, 2019(1), 2019.

[18] B. D. Lucas, T. Kanade, et al. An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th international joint conference on Artificial intelligence*, volume 2. IJCAI, 1981.

[19] R. C. Luo, C. T. Liao, K. L. Su, and K. C. Lin. Automatic docking and recharging system for autonomous security robot. In *2005 IEEE/RSJ Inter-*

national Conference on Intelligent Robots and Systems. IEEE, Aug 2005.

[20] C. McCarthy, N. Barnes, and R. Mahony. A robust docking strategy for a mobile robot using flow field divergence. *IEEE Transactions on Robotics*, 24(4), Aug 2008.

[21] M. A. Mehralian and M. Soryani. Ekfpnp: Extended kalman filter for camera pose estimation in a sequence of images. *arXiv preprint arXiv:1906.10324*, 2019.

[22] J. Moras, V. Cherfaoui, and P. Bonnifait. A lidar perception scheme for intelligent vehicle navigation. In *2010 11th International Conference on Control Automation Robotics & Vision*, pages 1809–1814. IEEE, 2010.

[23] M. Oberweger, P. Wohlhart, and V. Lepetit. Hands deep in deep learning for hand pose estimation. *arXiv preprint arXiv:1502.06807*, 2015.

[24] G. Pavlakos, X. Zhou, A. Chan, K. G. Derpanis, and K. Daniilidis. 6-dof object pose from semantic keypoints. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017.

[25] M. Persson and K. Nordberg. Lambda twist: an accurate fast robust perspective three point (p3p) solver. In *Proceedings of the European Conference on Computer Vision*. ECCV, 2018.

[26] S. Ren, K. He, R. B. Girshick, and J. Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *CoRR*, abs/1506.01497, 2015.

[27] F. J. Romero-Ramirez, R. Muñoz-Salinas, and R. Medina-Carnicer. Speeded up detection of squared fiducial markers. *Image and vision Computing*, 76, 2018.

[28] J. Röwekämper, C. Sprunk, G. D. Tipaldi, C. Stachniss, P. Pfaff, and W. Burgard. On the position accuracy of mobile robot localization based on particle filters combined with scan matching. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3158–3164. IEEE, 2012.

[29] G. Schweighofer and A. Pinz. Globally optimal o (n) solution to the pnp problem for general camera models. In *Proceedings of the 2008 British Machine Vision Conference*. BMVC, 2008.

[30] E. Shalnov and A. Konushin. Convolutional neural network for camera pose estimation from object detections. *International Archives of the Photogrammetry, Remote Sensing & Spatial Information Sciences*, 42, 2017.

[31] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza. *Autonomous mobile robots*. MIT press, 2011.

[32] S. Thrun, W. Burgard, and D. Fox. *Probabilistic robotics*. MIT Press, 2002.

[33] B. Triggs. Camera pose and calibration from 4 or 5 known 3d points. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 1, pages 278–284. IEEE, 1999.

[34] E. Yurtsever, J. Lambert, A. Carballo, and K. Takeda. A survey of autonomous driving: Common practices and emerging technologies. *arXiv preprint arXiv:1906.05113*, 2019.

[35] X. Zhou, D. Wang, and P. Krähenbühl. Objects as points. *CoRR*, abs/1904.07850, 2019.

Autonomous Grasping of Known Objects Using Depth Data and the PCA

Dominik Steigl, Mohamed Aburaia, Wilfried Wöber
UAS Technikum Wien
{mr18m007,mohamed.aburaia,wilfried.woeber}@technikum-wien.at

Abstract. *Two main goals for automated object manipulation processes are cost reduction and flexibility. Time-consuming, costly object-specific fixtures can be replaced by vision systems, whereby the manipulators are extended with cameras so that multiple objects in the environment can be precisely identified. To be able to manipulate an object, it must be recognized first in the world, and then the pose must be calculated. Neural network approaches recognize and estimate the pose of an object in a single step and yield superior results, but rely on vast amounts of training data. This work describes an approach for estimating the pose of identified objects without pre-trained pose data. Template matching is used to recognize objects in depth images, and the pose is estimated through principal component analysis (PCA). The input to the algorithm is reduced to the template. Pre-existing knowledge about the object further improves accuracy. A maximum deviation of 0.2 cm from the ground truth has been achieved, which suffices for the industrial grasping task. The system was evaluated with real measurements taken with an RGB-D camera. This work resembles a first step to estimate an object's pose with linear statistical methods.*

1. INTRODUCTION

Industrial robots are efficient at picking up objects in a predefined, structured environment [10]. When mobile manipulators are deployed in a factory setting and costly fixtures have to be avoided, robots need the ability to identify and locate objects for manipulation. To overcome this problem, a vision system can be used. One way to give robot vision is to use two-dimensional images with depth information, also known as 2.5D images or RGB-D images. RGB-D images can be used to find and localize objects by analyzing the environment. Building on top of the recognized and classified object, pose estimation tries to estimate the six degrees of freedom (DOF) pose of an object in an image. For mobile manipulation of objects this information is needed to accurately grasp objects with a manipulator in the correct position and orientation.

The current state of the art approaches towards object recognition and pose estimation are based on deep neural networks [15]. They usually outperform human crafted features [19], but unfortunately they rely on huge amounts of training data for classification and pose estimation and are difficult to adapt [9]. This is why, in this work, a more traditional approach was chosen. The target object is recognized using template matching in a 3D space. Pose estimation is implemented using the principal component analysis (PCA) to place an orthogonal basis in the center of the grabbing area. Using PCA to estimate the pose of the object, the needed input to the algorithm can be reduced to only the template. This work resembles a first step to estimate an object's pose with linear statistical methods.

In the following chapters the related work is summarized, the used methods are explained and the results are being discussed.

2. RELATED WORK

Object recognition describes the task of localizing known objects in images. Due to changes in the viewpoint or lighting, the task of mapping the huge amount of input pixels to a small output space is still complex [16]. To mitigate the influence of lighting conditions, approaches which rely on 3D information were researched [8]. The data used in these approaches is usually made up of a three channel 8-bit RBG image or an additional fourth channel which represents the 3D distance of the object to the image sensor, where each image is described using

$$I \in \mathbb{R}^{rows\ x\ cols\ x\ channels} \quad (1)$$

While research improved object recognition and classification with deep neural nets [15] , parallel efforts focused on template matching for object recognition [2][5]. Template matching uses extracted example images to find objects in new images. This method often involves sliding-window based algorithms [7], which find the template in a rectangular subpart of the image. Template matching works well for frontal images, but fails if the viewpoint differs from the actual template [4]. The simplicity of this technique still inspired new research, which is why its performance has improved significantly over the last 10 years [6][11].

2.1. Pose Estimation

Building on top of the recognized object, it is possible to estimate the pose of the object relative to the camera. This process is called pose estimation and it consists of three general categories. In the first category, the object's pose is stored alongside its feature vectors. Consequently, each different observed orientation represents a separate detection, which results in automatically knowing the objects pose if the object is matched with a previously trained one.

The second category uses statistical techniques to align two given RGB-D images with each other. For this, Iterative Closest Point [3], or ICP, is the most commonly used algorithm and many variants exist for different applications [12][14].

The third category tries to combine the pose estimation step with the recognition process itself. This makes sense, since, as stated earlier, a different viewpoint can change the appearance of an object entirely. This category has been covered by recent research due to the emerging field of machine learning [18].

Unfortunately, all of the before shown methods need either vast amounts of training data or an accurate model of the object that has to be detected. In this work, a different approach is taken. The principal component analysis (PCA) [1] is used for estimating the pose of a known object. PCA's intended purpose is to extract principal components and reduce dimensionality between the input and the output space. Using PCA to estimate the pose of the object, the needed input to the algorithm can be reduced to only the template. The proposed process of pose estimation with PCA is shown in the next chapter.

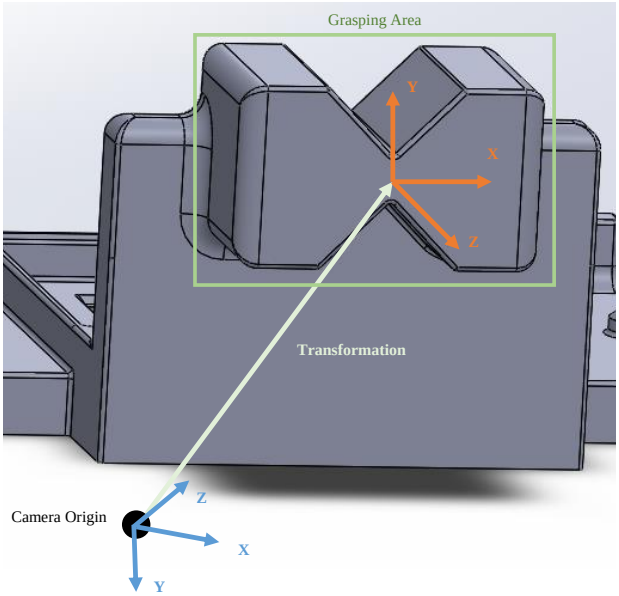


Figure 1. Visualization of the grasping point. The figure shows the object that has to be grasped. The orange coordinate system shows the center of the grasping area.

3. METHODS

The objective of the proposed approach is the estimation of the pose of a known object. Before the pose of an object can be calculated, it first has to be located in an image. For this task, template matching was chosen due to its ease of implementation and use. After the object has been recognized in the depth image, principal component analysis is used to determine the orientation of the found subpart of the image in 3D space.

Figure 1 shows the target object of this work. The pose of the shape in the "grabbing area" has to be calculated so that it can be successfully grasped. For this, the normal vector of the surface facing the camera has to be found. Through orientation of the vectors the rotational components of the 6D pose can be determined. This task can be solved by computing the PCA for the points in the grabbing area. In this case, the principal component analysis yields 3 eigenvectors with their respective eigenvalues for the given 3D points. As can be seen by studying Figure 1, 2 of the 3 dimensions of the shape in the grabbing area differ from the other. The span of values in the X and Y direction are comparatively large in respect to the depth dimension Z. This also applies to the respective variances. Using prior knowledge, the normal vector of the plane parallel to the camera origin (i.e. corresponding to the surface of the marked grabbing area) can be estimated using the eigenvec-

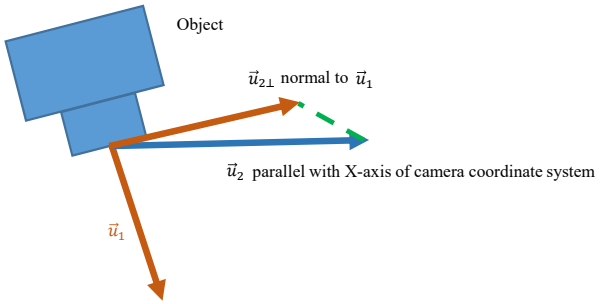


Figure 2. Visualization of the correct alignment of vector $\vec{u_2}$

tor with the smallest eigenvalue (eg. variance).

As the form of the shape is symmetrical, the mean of the points in the grabbing area estimates the origin of the coordinate system shown in Figure 1. Therefore, the mean of the PCA can be used as the translational component of the transformation matrix.

$$\vec{t} = (\vec{\mu_x},\ \vec{\mu_y},\ \vec{\mu_z})^T \tag{2}$$

The rotation matrix has to be assembled from three orthonormal vectors. The first vector has already been found, which is the smallest eigenvector of the PCA, which forms the Z vector pictured in Figure 1. The second vector can be obtained by leveraging knowledge about the environment of the industrial grasping use case. As the target object is located at a target location that is parallel to the ground, the rotation around the Z axis can be neglected. That is why the second vector can be aligned with the Y axis of the camera coordinate system. But since the first vector found with the PCA could be rotated around the Y axis of the object coordinate system, the second vector has to be projected orthogonally to the first. This is done with Equation (3) and the process is visualized in Figure 2.

$$\vec{u_{2||}} = (\vec{u_2^T} \cdot \vec{u_1}) \cdot \vec{u_1} \tag{3}$$
$$\vec{u_{2\perp}} = \vec{u_2} - \vec{u_{2||}}$$

The third vector can then be calculated using the cross product of $\vec{u_1}$ and $\vec{u_2}$. The resulting rotation matrix is constructed using Equation (4).

$$\mathbf{R} = \begin{pmatrix} u_{3,x} & u_{3,y} & u_{3,z} \\ u_{2,x} & u_{2,y} & u_{2,z} \\ u_{1,x} & u_{1,y} & u_{1,z} \end{pmatrix} \tag{4}$$

After calculating the rotation and translation components of the object, a transformation matrix can be formulated using Equation (5).

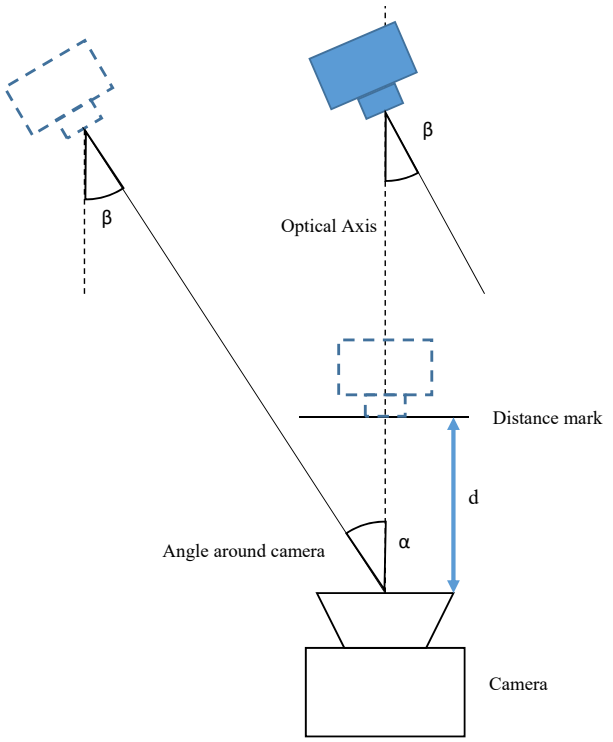


Figure 3. Visualization of the test setup

$$T_{obj}^{camera} = [\ \mathbf{R} \quad \vec{t}\ ] \tag{5}$$

The transformation matrix can then be used to express the grasping point in the world coordinate system, which is used for motion planning of the robot arm. Having calculated the transformation between the camera coordinate system and the objects coordinate system one can calculate the objects world position as follows

$$T_{obj} = T_{camera}^{world} \cdot T_{obj}^{camera} \tag{6}$$

In the following chapter, the performance of the proposed approach is discussed.

4. Results

The test setup consisted of the 3D printed model of the target object shown in Figure 1 and an Intel RealSense D435[1]. The RGB-D camera has been set up at a defined location on a table and the 3D printed model has been placed in front of it as can be seen in Figure 3.

To measure the error of the PCA-based approach, a metric had to be defined. For this, the Euclidean distance between the ground truth vector and the estimated plane normal vector of the PCA is used. Usu-

[1] https://www.intelrealsense.com/depth-camera-d435/

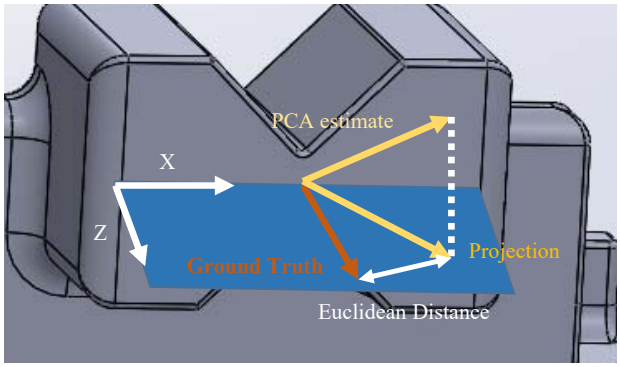


Figure 4. Visualization of calculation of the Euclidean distance between the ground truth vector and the vector estimated by the PCA

ally, with machine learning approaches, the error in the 2D projection of the 3D bounding box is measured [13]. Since the estimation error can be measured directly in this case, the Euclidean distance is used as a metric instead. To ease the calculation of the ground truth vector, environment knowledge has been used to eliminate one dimension out of the 3D vector. Since the target 3D model is guaranteed to always be parallel to the ground, as is the camera, the rotation around the Z-axis defined in Figure 1 can be ignored. Furthermore, as this approach is being used in an industrial grasping use case where the industrial robot has to grab the target object perpendicular to the estimated plane, the Y-component of the estimated PCA vector can be ignored and therefore set to 0. In order to get two vectors of the same length for further correct calculation, both, the ground truth vector and the vector estimated by the PCA have to be normalized. This results in an Euclidean distance being calculated between two vectors in the X-Z plane. This process is shown in Figure 4.

Equation (7) shows the calculation of the ground truth vector, where the $\vec{gt}$ vector is the ground truth. The X and Z components of the ground truth vector can be obtained by calculating the direction of the ground truth vector rotated by β depicted in Figure 3.

$$\vec{gt} = \begin{pmatrix} \sin(\beta) \\ 0 \\ \cos(\beta) \end{pmatrix} \tag{7}$$

Equation (8) shows the calculation of the error in form of the Euclidean distance.

$$r = \sqrt{(x_1 - x_2)^2 + (z_1 - z_2)^2} \tag{8}$$

x_1 and z_1 denote the respective components of the ground truth vector. x_2 and z_2 denote the respective components of the calculated normal vector by the PCA, that has been projected onto the X-Z plane.

Table 1. List of positions that were used for the experiments.

Angle [°]	Distance [cm]
+/- 0	30, 35, 40, 45, 50, 75
+/- 10	
+/- 20	
+/- 40	30, 35, 40, 45, 50
+/- 10 around camera	30, 35, 40, 45, 50, 75
+/- 20 around camera	
+/- 30 around camera	

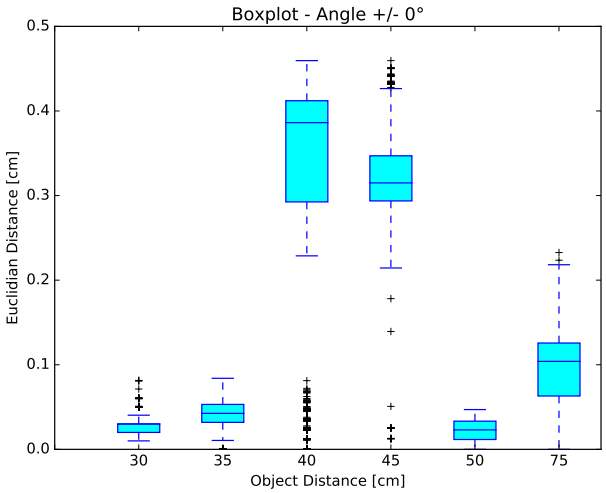


Figure 5. Visualization of the results with the object being placed on the optical axis

The measurements were taken in distances and orientations that relate to the industrial grasping use case. The target object has been moved to several fixed positions in front of the camera. Table 1 lists the positions that were used for the measurements.

Figure 5 shows the results for measurements taken with the object being placed on the optical axis.

Both of the anomalies at 40 and 45cm can be explained due to poorly selected templates. This can be mitigated by using advanced approaches for template matching [5][17]. Those rely on scaling of the template to get a more accurate match and also address the rotational limitations.

Figure 7 shows an example disparity image of the object viewed by the Intel RealSense camera. It can be argumented that the anomalies are induced because of the dark areas in the disparity image, which can be mostly traced back to occlusions of the stereo vision system. This has an even larger effect when

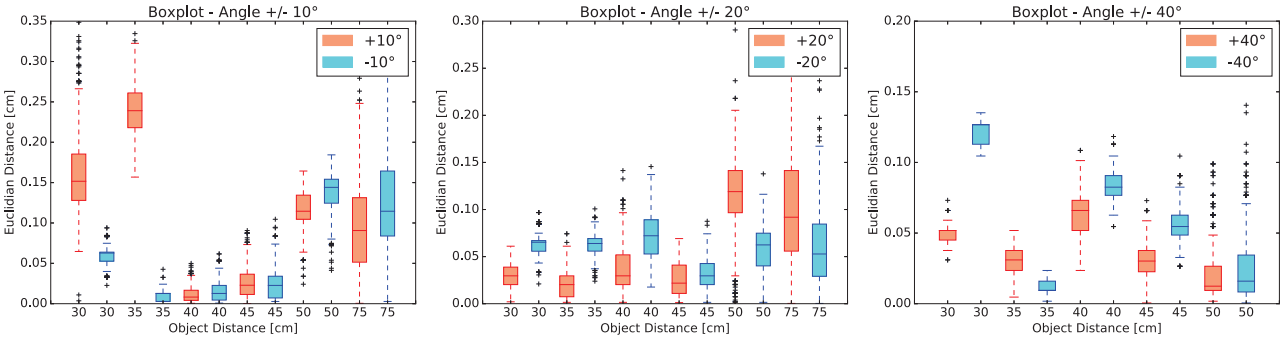


Figure 6. Visualization of the results with the object being placed at differing angles and distances on the optical axis

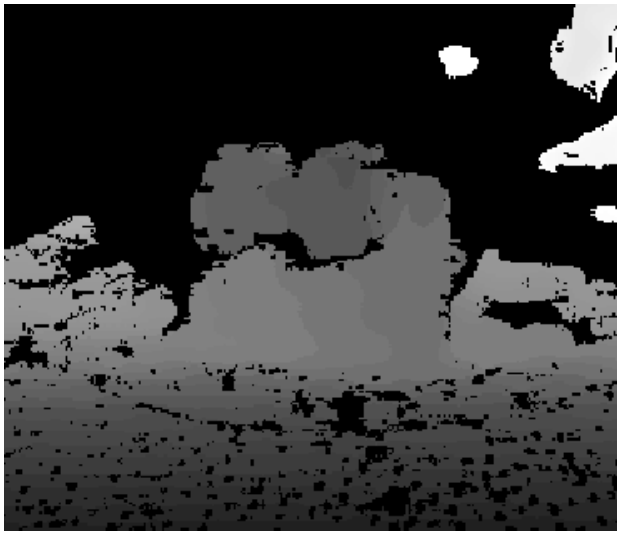


Figure 7. Disparity image of the object viewed by the Intel RealSense camera. Occlusions induced by the stereo vision system make it difficult to accurately locate the grasping area depicted in Figure 1.

the object is being rotated. Figure 7 also shows that it is difficult to depict the grabbing area of the object for creating a fitting template from the disparity image. Having a poorly chosen template leads to points being incorporated into the PCA estimation that are not actually part of the grabbing area and therefore lead to unexpected results. Nevertheless, it can be concluded that the anomalies are not induced by means of the method used for estimating the pose.

Figure 6 shows the results for measurements taken with the object being placed at different angles on the optical axis. Refer to Figure 3 for a visual presentation of this process. The angle depicted in Figure 6 corresponds to angle α shown in Figure 3. The distance mark relates to the distances shown on the X-axis labels of the graphs in Figure 6. The anomalies again can be explained by the problems mentioned before. The right graph in Figure 6 clearly shows the limits of the proposed approach, as the structure of the box plots over the graph changes in respect to the other two graphs.

Table 2. Regions in which the algorithm yields results sufficient enough for the industrial grasping use case at hand.

Angle [°]	Distance [cm]
+/- 0	30 - 75
+/- 10	
+/- 20	
+/- 10 around camera	30 - 75
+/- 20 around camera	
+/- 30 around camera	

Figure 8 shows the results for measurements taken with the object being placed at different angles around the camera. The angle depicted in Figure 8 corresponds to angle β shown in Figure 3. The anomalies again can be explained by the problems mentioned before.

The results show that the usable region for this algorithm can be summarized with Table 2. Argumenting, that the anomalies can be eliminated by using the enhancements already listed. Angles depicted with the ”around camera” suffix correspond to the object being rotated around the camera with angle α, as depicted in Figure 3.

5. Conclusion

This work presented an approach to estimate the pose of a known object by using the principal component analysis. This resembles a first step to estimate an object's pose with linear statistical methods. The results showed that the approach is sufficient for the industrial use case at hand, since a maximum deviation of 0.2 cm compared to the ground truth is achieved, when anomalies are ignored. The results also show the limitations of this approach. Anoma-

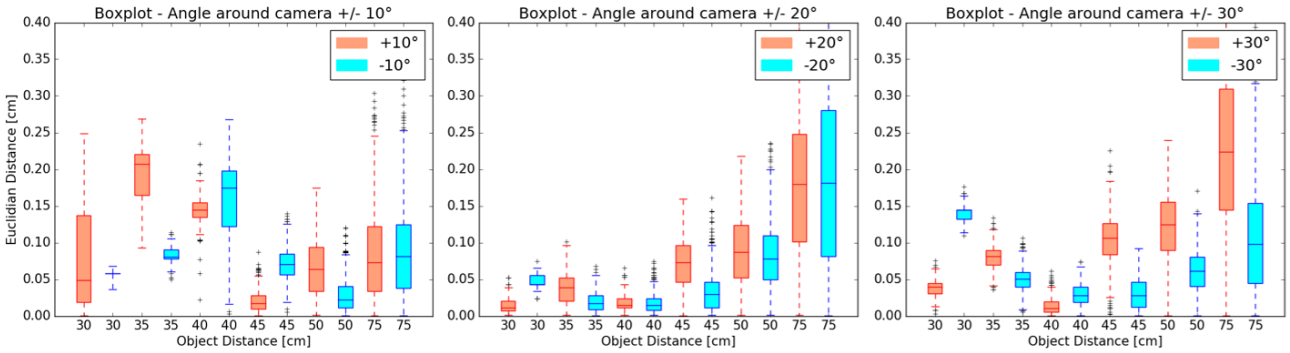


Figure 8. Visualization of the results with the object being placed at different angles and distances around the camera

lies shown in the data can be explained through poorly chosen templates. The problems faced could be solved in future work by using the recommendations given in this work.

References

[1] H. Abdi and L. J. Williams. Principal component analysis. *WIREs Computational Statistics*, 2(4):433–459, 2010.

[2] D. I. Barnea and H. F. Silverman. A Class of Algorithms for Fast Digital Image Registration. *IEEE Transactions on Computers*, C-21(2):179–186, Feb. 1972.

[3] P. Besl and N. D. McKay. A method for registration of 3-D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, Feb. 1992.

[4] L. Cole, D. Austin, and L. Cole. Visual Object Recognition using Template Matching. In *Proceedings of Australian Conference on Robotics and Automation*, 2004.

[5] R. M. Dufour, E. L. Miller, and N. P. Galatsanos. Template matching based object recognition with unknown geometric parameters. *IEEE Transactions on Image Processing*, 11(12):1385–1396, 2002.

[6] S. Korman, D. Reichman, G. Tsur, and S. Avidan. Fast-Match: Fast Affine Template Matching. *International Journal of Computer Vision*, 121(1):111–125, Jan. 2017.

[7] K. Lai, L. Bo, X. Ren, and D. Fox. A large-scale hierarchical multi-view rgb-d object dataset. In *2011 IEEE international conference on robotics and automation*, pages 1817–1824. IEEE, 2011.

[8] Y. Lu and D. Song. Robustness to lighting variations: An RGB-D indoor visual odometry using line segments. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 688–694. IEEE, 2015.

[9] G. Marcus. Deep learning: A critical appraisal. *arXiv preprint arXiv:1801.00631*, 2018.

[10] V. Nabat, M. de la O RODRIGUEZ, O. Company, S. Krut, and F. Pierrot. Par4: very high speed parallel robot for pick-and-place. In *2005 IEEE/RSJ International conference on intelligent robots and systems*, pages 553–558. IEEE, 2005.

[11] R. Opromolla, G. Fasano, G. Rufino, and M. Grassi. A model-based 3d template matching technique for pose acquisition of an uncooperative space object. *Sensors*, 15(3):6360–6382, 2015.

[12] F. Pomerleau, F. Colas, R. Siegwart, and S. Magnenat. Comparing ICP variants on real-world data sets. *Autonomous Robots*, 34(3):133–148, Apr. 2013.

[13] J. N. Rauer. Semi-Automatic Generation of Training Data for Neural Networks for 6d Pose Estimation and Robotic Grasping. Master's thesis, Fachhochschule Technikum Wien, Höchstädtplatz 5, 1200 Wien, 2019.

[14] S. Rusinkiewicz and M. Levoy. Efficient variants of the ICP algorithm. In *Proceedings Third International Conference on 3-D Digital Imaging and Modeling*, pages 145–152, May 2001. ISSN: null.

[15] C. Szegedy, A. Toshev, and D. Erhan. Deep neural networks for object detection. In *Advances in neural information processing systems*, pages 2553–2561, 2013.

[16] B. S. Tjan and G. E. Legge. The viewpoint complexity of an object-recognition task. *Vision Research*, 38(15):2335–2350, Aug. 1998.

[17] F. Ullah and S. Kaneko. Using orientation codes for rotation-invariant template matching. *Pattern recognition*, 37(2):201–209, 2004.

[18] P. Wohlhart and V. Lepetit. Learning Descriptors for Object Recognition and 3d Pose Estimation. *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Feb. 2015.

[19] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox. PoseCNN: A Convolutional Neural Network for 6d Object Pose Estimation in Cluttered Scenes. *arXiv:1711.00199 [cs]*, May 2018. arXiv: 1711.00199.

EDLRIS: European Driving License for Robots and Intelligent Systems

Manuel Menzinger, Martin Kandlhofer, Gerald Steinbauer
Graz University of Technology
{menzinger,kandlhofer,steinbauer}@ist.tugraz.at

Ronald Bieber, Wilfried Baumann, Margit Ehardt-Schmiederer
Austrian Computer Society OCG
{ronald.bieber,wilfried.baumann,margit.ehardt-schmiederer}@ocg.at

Thomas Winkler
University of Teacher Education Burgenland
thomas.winkler@virtuelle-ph.at

Abstract. *EDLRIS is a professional and standardized system for training and certifying people in fundamental topics of Robotics and Artificial Intelligence. It was developed, implemented and evaluated within the course of an international 3-year project. This paper provides an overview of goals, methodology, training modules and preliminary results of the EDLRIS project.*

1. Introduction

Robotics and Artificial Intelligence (AI) have a big impact on the working world and on people's everyday life. An increasing number of jobs are related to Robotics and AI, resulting in a strong demand for well-trained people in these areas. In order to foster a solid understanding of sociopolitical, economical and technical aspects it is important to teach fundamental Robotics and AI concepts already prior to, or outside of university. Nevertheless, hardly any well-founded teaching approaches exist at the moment. In order to address this challenge, the *European Driving License for Robots and Intelligent Systems (EDLRIS)* was developed. It specifically focuses on teaching fundamental concepts of Robotics and AI to trainers (e.g. educators, teachers, mentors, ...) and trainees (e.g. young people, pupils, apprentices, ...) following a train-the-trainer, blended-learning approach [1].

2. Related Work

The project idea was inspired by the *European Computer Driving License (ECDL)* [2]. A lot of Robotics and AI courses are held at undergraduate or graduate level (e.g. [3]) but training and certifying people in fundamental topics of Robotics/AI outside university hardly exists. Several pre-university approaches teach only selected or very basic topics of Robotics/AI (e.g. [4]). In recent years, education organizations started to develop AI curricula and programs for a K-12 audience (e.g. Elements of AI [5]). However, training and certifying trainers as well as young people in fundamental Robotics/AI topics, combining face-to-face and online teaching units - as done by EDLRIS - is quite unique.

3. Methodology

The general approach of EDLRIS is based upon following main stages: **1) preparation:** conducting a pre-survey among stakeholders and establishing an advisory board with representatives from industry and education; **2) development:** developing Robotics and AI training modules including a certification system to prove the acquired skills of trainers and trainees; **3) train the trainers:** conducting training courses and certifications for trainers (face-to-face (f2f) and online teaching units); **4) train the trainees:** educating and certifying trainees by certified trainers who act as multipliers;

EDLRIS comprises 4 modules: *Robotics Basic/Advanced* and *AI Basic/Advanced*. All modules have a strong focus on hands-on activities and include practical tasks based on the principles of constructionism [6]. *Basic* modules focus on people without any prior knowledge, aiming at building

awareness, motivating and introducing fundamental concepts in an easily comprehensible manner (scope: 24 hours f2f, 20 hours online). *Advanced* modules primarily focus on people who already have prior knowledge in computer science/mathematics, aiming at enabling a deeper understanding of fundamental concepts (scope: 36 hours f2f, 50 hours online).

Exemplary, the following gives an insight into the **Robotics Advanced** module, which puts the focus on a fundamental understanding of robotic arms and mobile robots [7]. Preparatory online sessions provide the necessary basics in calculus, linear algebra and Python programming. During the subsequent f2f units, participants are given two concrete problems: **1)** mathematical description/modelling of a certain robotic arm and its trajectory; **2)** indoor localization and navigation of a mobile robot; By working on these tasks, participants learn about the kinematical model (direct/inverse kinematics, homogeneous transformation, DOF/DOM, Jacobian) as well as sensor fusion and state estimation (probabilistic model, Bayesian and Kalman filter). Teaching tools are paper+pencil exercises, simulators (Python) and the *TurtleBot 2* robotics platform. The module concludes with the final exam (certification).

A detailed description of all modules can be found at [1] and on the project website (edlris.eu).

4. Implementation and Evaluation

In 2019, 19 Robotics and AI training courses were conducted and evaluated using quantitative and qualitative methods. In sum, 271 people participated, whereas 66% also successfully completed the certification. The majority (76%) of participants were trainers. A survey among participants was administered prior and after each course (Likert scale, open-ended questions). Summing up the results, 92% stated that their expectations towards the training were met and over 90% that the face-to-face (f2f) units were essential for their learning success. On the contrary, only 80% agreed that the online units were sufficiently aligned with the f2f units[1]. Furthermore, participants mentioned that the gaps (in terms of complexity) between the *Basic* and *Advanced* modules are too large, making it hard for young trainees to fully understand the complex, advanced topics. In addition to the survey, quantitative pre- and post-tests at *AI Basic* trainings were conducted using a questionnaire with 10 multiple-choice knowledge questions. Data analysis (paired t-test) showed a statistically significant learning gain ($t(21)=18.086$, $p<.001$). Further data analysis is ongoing and more extensive pre-/post-test evaluations will be conducted during the upcoming training courses.

[1] average percentage over all 4 training modules

5. Conclusions and Future work

This paper presented the *European Driving License for Robots and Intelligent Systems (EDLRIS)*, a training and certification system to teach people fundamental concepts of Robotics and AI. The first training courses have been implemented and evaluated in Austria and Hungary in 2019, and, due to the great demand, further trainings and certifications will be conducted in 2020. In order to get a better founded assertion regarding the success of the entire system, a more extensive quantitative evaluation will be implemented. Furthermore, contents and structure of the training modules will be adapted according to insights and lessons-learned from the first implementations.

Acknowledgements

This project is supported by the European Union funding programme Interreg V-A AT-HU 2014-2020

References

[1] M. Kandlhofer, G. Steinbauer, J. Lassnig, W. Baumann, S. Plomer, A. Ballagi, and I. Alfoldi, "Enabling the creation of intelligent things," in *IEEE Frontiers in Education Conference (FIE)*, 2019.

[2] N. Csapo, "Certification of computer literacy," *THE journal*, vol. 30, no. 1, 2002.

[3] C. N. Silla, M. Paglioney, and I. G. Mardegany, "jothellot: A java-based open source othello framework for artificial intelligence undergraduate classes," in *IEEE Frontiers in Education Conference (FIE)*, 2016.

[4] C. A. Heinze, J. Haase, and H. Higgins, "An action research report from a multi-year approach to teaching artificial intelligence at the K-6 level," in *Symposium on Educational Advances in Artificial Intelligence*, 2010.

[5] ElementsOfAI, "Elements of AI online course," 2019. accessed April 9, 2020.

[6] S. Papert, *Mindstorms: Children, computers, and powerful ideas*. Basic Books, Inc., 1980.

[7] B. Siciliano and O. Khatib, *Springer handbook of robotics*. Springer, 2016.

Design and Implementation of a Mobile Search and Rescue Robot

Georg A. Novotny and Wilfried Kubinger
UAS Technikum Vienna, Hoechstaedtplatz 6, 1200 Vienna
{georg.novotny,wilfried.kubinger}@technikum-wien.at

Abstract. *For public emergencies such as nuclear accidents or natural disasters, an urgent and reliable description as well as an evaluation of the environment form the basis of all organized search and rescue (S&R) team plans and actions. If this information is not available the risks for the rescue services increases dramatically. Mobile robots help to minimize these risks by providing information about the disaster site to rescue teams.*
This paper discusses the needs and requirements of mobile robots in S&R application areas such as nuclear disasters and evaluates results achieved during the ENRICH 2019 trial based on the system architecture of the mobile S&R robot "Robbie" of UAS Technikum Vienna. The successful participation of the ENRICH 2019 show that the mobile robot is capable of performing S&R actions during emergencies.

1. INTRODUCTION

One of the main reasons for deaths after disasters is that it takes rescue teams too long to discover victims because they need to ensure their own safety [13, 22]. Rescue robots are designed for situations like these that are too dangerous for humans, e.g. hostage taking, nuclear or natural disasters [21]. S&R robots eliminate the need for human scouts to expose themselves in hazardous environments by creating an awareness of the situation at the disaster site by providing immediate feedback to the rescue workers before they enter the disaster site [22]. To support the development of rescue robots, since the beginning of the 2000s a number of robot trials and competitions have been held, such as: "*ELROB*" - The European Land Robot Trail , "*EnRicH*" – European Robotics Hackathon [8], the "*Arctic Robot Challenge*" [1], "*Rescue Robot League*" [27], the "*DARPA*" – Defence Advanced Research Projects Agency [6], the "*EuRoC*" – European Robotics Challenges [10] or the "*EU-FP7-ICARUS*" [12] project and many other. During these trials different tasks need to be solved in various environments, ranging from 2D and 3D mapping to the detection and evacuation of people and manipulation of objects [8, 9, 1, 27, 6, 10, 12].
The remainder of this chapter the requirements for mechanical design, sensor configuration and graphical user interface (GUI) for S&R robots are evaluated, followed by the evaluation of the implemented hardware and the developed software of the FH Technikum Vienna in section 2. The results achieved with these setup are highlighted and discussed in section 3. Concluding section 4 summarizes this work and gives an overview of future work.
Table 1 summarizes the user requirements for S&R robots collected in [7] and [24]. Disaster areas are

Requirements for Search and Rescue Robots	
Topic	**Requirement**
Dimensions	The robot platform must fit on 2 standard Euro pallets (120cm x 160cm x 95cm) and must not weigh more than 100kg
Nr. of Operators	Two people must be enough to operate the S&R robot
Resistance	IP65 for outdoor unmanned ground vehicles (UGV)
Autonomy	Must be possible to immediately switch from autonomous to tele-operated
Sensing	Video (RGB and/or thermal) cameras for visual contact with victims, 3D sensors to generate a structural map of the environment
Communication	Connection loses will occur → ad-hoc networks required
Command and Control	Simple interfacing technologies only on high-level tasks
Ambient Light	Must be capable of working in complete darkness as well as light environments
Energy Requirements	Energy consumption should be lower than 2kVA for recharging
Graphical User Interface	Camera view(s) from robot's perspective + environmental perceptions
	Sensor and status information of initial state and sensors
	Bird's eye view map

Table 1. Summary of system requirements survey, data taken from [7, 24]

usually covered with rubble and debris and can extend over several floors. Therefore the base platform of the mobile robot must be able to manoeuvre in rough terrain and should be suitable for climbing stairs. As stated in [22] track based robots are designed for operation in uneven, debris-covered terrain and are therefore ideally suited for natural disasters, moreover these tank like tracks add stability to the whole robot system [25]. Manipulation of objects is also often required by S&R robots, whether for demining, interaction with victims or for generating an unique camera angle [24].
Now that the system requirements have been defined, following chapter examines the approach of the UAS to integrate these requirements into a S&R robot.

2. System Concept

Following section explains the hardware and sensor setup of the robot of the UAS as well as the software architecture for successful participation in the EnRicH 2019 trial or other S&R applications.

2.1. System design

Figure 1 gives an overview of the implemented hardware components which are described in detail in the next section. As visualized the setup consists

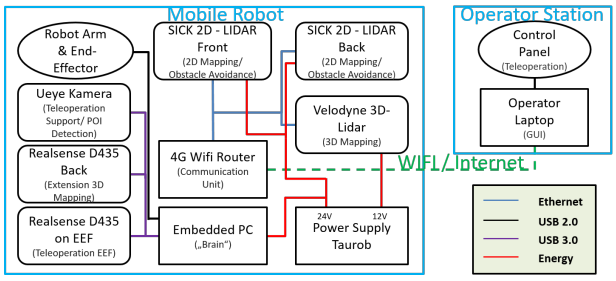


Figure 1. System design overview

of two main parts, the "*Mobile Robot*" and the "*Operator Station*". Although mobile S&R robots have autonomous capabilities an operator station is needed to provide a safety fallback and teleoperation system. The mobile robot itself needs to be equipped with numerous sensors ranging from 2D and 3D LIDARs for obstacle avoidance and mapping, an robotic arm including an end effector (EEF) for manipulation and front and rear facing cameras for teleoperation.
The remainder of this chapter describes implemented hard- and software needed to provide a mobile S&R robot.

2.2. Implementation - Hardware

The track steered mobile robot "*Tracker*" of company taurob GmbH [31] is used as the basic building block. Due to adjustable crawler tracks, a high degree of off-road mobility is provided for maximum versatility [31]. In addition a 4 degrees of freedom (DOF) robot arm, of taurob GmbH, for manipulation tasks was mounted. The Robot Operating System (ROS) API provided by taurob GmbH was the decisive factor for this mobile robot platform. The "*Tracker*", manipulator and the current sensor setup are depicted in figure 2.

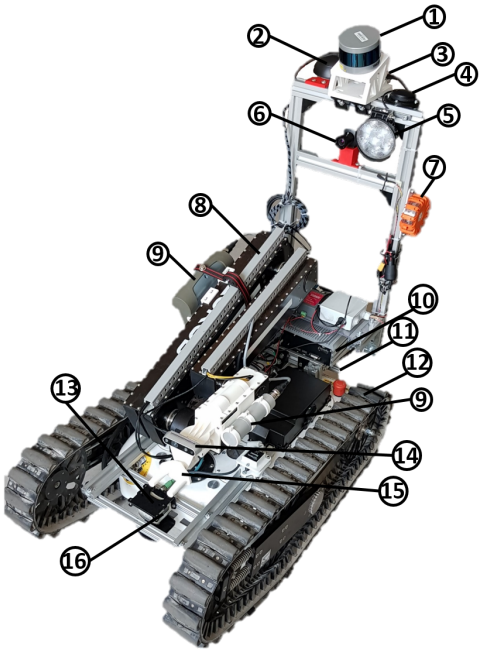


Figure 2. Robbie hardware setup
(1) Velodyne PUCK-VLP16, (2) Wifi/ 4G Antenna, (3) Rear-Facing Intel Realsense D435, (4) Garmin GPS Module, (5) LED Headlight, (6) Ueye UI-3240LE Camera with Camera Mount, (7) Operation Indication Light, (8) Robotic Arm, (9) SSM1+ Radiatiometer with mounted Probe, (10) Embedded PC, (11) Rear-Facing Internal Camera and SICK TIM-551-2050001, (12) taurob Tracker, (13) Front-facing TIM-551-2050001, (14) Intel Realsense D435 mounted on EEF, (15) EEF, (16) Front-Facing Internal Camera and Internal Headlights

To allow a maximum level of flexibility a modular hardware setup consiting of a sensor rig was chosen, thus enabeling easy replacement of sensors as well as software to enable different S&R tasks. A Garmin GPS module was attached to the sensor rig for outdoor localization. For GPS-limited indoor scenarios depth and range sensors were used for localization and mapping. Therefore, a 3D Light Detection and Ranging (LIDAR) Velodyne PUCK VLP-16 [32] and two SICK TIM-551-2050001 [30] 2D LIDARs, one at the front and one at the rear, were attached to the robot. In contrast to the 2D LIDARs mounted in a planar arrangement, the 3D LIDAR was mounted at an angle of 20° to Robbie's direction of travel.

To enable accurate tele-operation, the LIDARs were used in combination with appropriate software (see section 2.4) to generate a 2D and 3D map of the environment, giving the user insights into the environment from Robbie's point of view (POV), as shown in figure 3 (8) (7). To include the environment behind Robbie, which cannot be captured by the 3D LIDAR an RGB-Depth (RGB-D) camera, the Intel D435 [18], has been mounted slightly facing downwards on the sensor tower of the mobile robot. An additional Intel D435 was mounted on the base plate of the EEF to facilitate tele-operation of the EEF and Image Based Visual Servoing (IBVS) [4]. In addition, a Phidgets Spatial Inertial Measurement Unit (IMU) was mounted on the base platform to improve localization using sensor fusion such as Extend Kalman Filters and to provide input for tip-over control [5, 20, 23]. To enable an elevated POV, a universal camera mount with an attached Ueye UI-3240LE [17] camera was mounted on the sensor rig. Finally, to enable radioactive and nuclear (RN) detection, the robot was equipped with a radiometer SSM1+ [29].

The processing of this sensor data is a computationally complex process, so an industrial computer with the following specifications was also installed on Robbie's base platform:

- 1 × Intel Core(TM) i7-7700T (4 Cores, 8 Threads) @ 2.90GHz
- 1 × GeForce GTX 1050 Ti
- 2 × 16GB DDR4 2133 MHz

The visualization of these different sensor readings is a difficult task. Therefore an intuitive GUI for Robbie was developed, which is discussed in the next section followed by the implemented software.

2.3. User Interface

The user interface is the essential component for promoting situational awareness [24]. To underline this statement, the reader's attention is drawn to the fact that an S&R robot was rejected in the tragedy of 9.11. because of a too complex user interface [24]. Figure 3 shows the operator station, of the UAS Technikum Vienna, with the associated user interface and control panel.

As depicted on the left hand side of figure 3 the user interface is split into three parts:

1. **Log-Screen / Command input** (Figure 3 (1))
   All log messages of the running software are displayed here, this is a necessity to detect software system errors. In addition, these terminal windows can be used to start/restart any software modules, this allows a maximum level of flexibility.

2. **GUI** (Figure 3 (2))
   The GUI allows the operator to perceive the environment from Robbie's POV, which is a necessity for S&R robots [24]. This is achieved by live streams from the cameras (8). In the default configuration, the internal, forward and backward facing cameras of the tracker and the elevated RGB camera are streamed. Furthermore, the 2D map generated by the SLAM approach discussed in section 2.4 is visualized in the middle part of the GUI as shown in (7), providing a bird's eye view for the operator. Finally, sensor values (4), such as the internal temperature, battery voltage and estimated time to shutdown and the detected emitted radiation are displayed in counts per second. The developed GUI thus visualizes all suggestions for a good user interface evaluated in [24], which further enables the optimization and interoperability of the available resources and accelerates access to the victims [7]. In addition, it is also possible to visualize additional sensor values and information by a simple mouse click (5), such as a 3D map, the additional camera on the gripper, the backwards

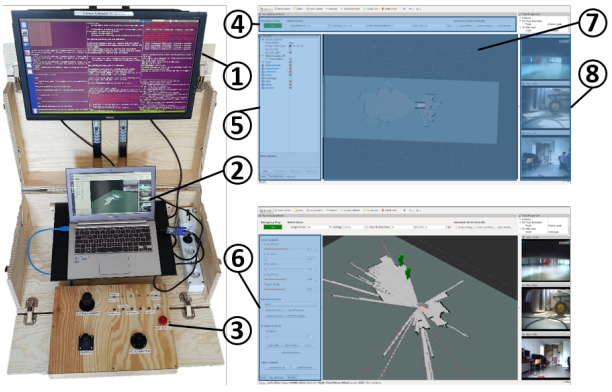


Figure 3. left: Complete operator station right: GUI (1) Logscreen / Command input, (2) GUI depicted in more detail on the right, (3) Control panel for teleoperation, (4) Sensor readings and emergency of switch, (5) Topic visualisation checkbox, (6) Additional teleoperation toolbox, (7) Map visualisation toolbox, (8) Image stream from Robbie's POV

facing camera, the autonomously detected people or the local and global cost map for the autonomous drive. In addition, the GUI can also be used for tele-operation of the robot arm in case the connection to the control panel fails, so that a redundant tele-operation system is available (6).

3. **Control panel** (Figure 3 (3))
   The control panel for teleoperation is used to tele-operator Robbie. Here the steering of the base platform as well as the robot-arm is handled. Further the autonomous "come-home" functionality can be started and stopped.

2.4. Implementation - Software

The Robot Operating System (ROS) [14] is used as high-level API to evaluate sensor data and control actuators. To improve the tele-operation process, a GUI plug-in for rviz [19] has been developed, which displays all sensor data and enables tele-operation of the robot arm (see figure 3).

For 2D and 3D mapping the open source frameworks Cartographer [15] and Octomap [16] were implemented. The main advantage of the Cartographer algorithm is the ability to detect and calculate online loop closure with graph optimization, which minimizes the absolute translation and rotation errors during map generation [15]. Octomap, on the other hand, uses a probabilistic estimation of the occupancy of 3D space and represents the environment in octaves, which consists of occupied voxels [16]. Figure 4 visualize a generated 2D or 3D map with these SLAM approaches using the LIDAR sensors listed in section 2.2, recorded during the EnRicH 2019 trial.

To overcome the need for manual victim recognition and mappinga ROS package based on Octomap and YOLO-ROS [2], a convolutional neural network (CNN) for object recognition in RGB images, was developed. By utilizing ray casting and the Bounding Box the x and y coordinates of the victim, with the 6 DOF transformation between the map frame and the RGB camera frame, is calculated. Detected victims are visualized on the 2D and 3D map of the GUI. By utilizing 2 different sensors to calculate the position of the victim thermal imaging cameras can also be used for victim detection.

Further a ROS package for automatic drive has been developed which uses the move-base-flex framework [26], a flexible navigation framework, and SMACH [3], a task level architecture for building complex robot behaviors. Currently two path planners are implemented. The Timed Elastic Band (TEB) planner, a planner that takes travel time into account. Movement is not calculated by the simulated forces within the virtual elastic band, but by optimizing the travel time and the path [28]. The TEB planner calculates several feasible paths and selects the fastest one. If the planner does not reach the target, recovery behaviours are called up. After each behaviour call the planner tries to reach the target again. If the target is still not reachable, the next recovery behaviour is called. After all three implemented behaviors are executed, the local scheduler is switched to the Dynamic Window Approach (DWA) algorithm. The DWA breaks up the global plan into smaller windows, whereby only the current and the next window are used to calculate the path [11]. The speeds within the next window are calculated using the current robot speed, the possible acceleration of the robot and objects to be avoided. The target tolerance of the DWA planner is increased to ensure that the target position can be reached. If the planner cannot reach the specified target, the recovery behaviours are called up as with the TEB planner. If the system still cannot reach the target after calling all recovery behaviors, the execution of the local and global planner is terminated. The SMACH script then returns an error and waits for a new target. The first implemented recovery behavior clears the cost maps, the second moves the robot back for 0.3m or 5 seconds and the third turns the robot 360° on the spot.

During the exploration the radioactivity is continuously measured with the radiometer. After exploration the nuclear radiation of the area around the driven path is estimated using a Gaussian process. The amount of radiation is then visualized and piled over the 2D map together with a legend, further the radioactivity is also visualized in the 3D map.
Section 3 now introduces the results achieved using this system concept during the 2019 EnRicH trial.

3. Results and Discussion

Table 2 evaluates the System Readiness Level (SRL) of the mobile Robot of the UAS Technikum Vienna based on the survey evaluated in [7, 24]. Using the survey results of [7] and [24] Robbie's SRL is defined as 9/10, since only the IP65 resistance could not be fulfilled.

Figure 4 visualizes the environment mapped dur-

Requirements for Search and Rescue Robots		
Topic	**Robbie**	**Status**
Dimensions	112×58×120cm, ca 75kg	☑
Nr. of Operators	Only one operator required	☑
Resistance	Currently no IP certificate due to active cooling	☒
Autonomy	Can be easily switched using GUI or control panel	☑
Sensing	Five image-streams provided, as well as 2D and 3D maps	☑
Communication	Automatic drive counters connectivity problems	☑
Command and Control	Intuitive GUI and operator station developed	☑
Ambient Light	Can be operated during day and night due to two LED headlights	☑
Energy Requirements	0.36kVA	☑
Graphical User Interface	Three image-streams on start-up, additional 2 can be started manually	
	Battery Voltage, Remaining operation time and sensor readings in status-bar	☑
	2D map on start-up, 3D map can be visualised manually	☑

Table 2. Evaluation of Robbie's applicability using the survey in [7] and [24]. For a description of the requirements see table 1.

ing the EnRicH 2019 as a 2D, 2D with marked radioactive sources and as a 3D map. As the 2D map (upper left) depicts, the calculated loop closure of the cartographer node distorted the map, resulting in sloping walls. This also led to the fact that two radioactive sources were merged into one by the radioactive mapping approach, since the odometry of the mobile robot is not recalculated during loop closure. As can be seen in figure 4, the generated 3D

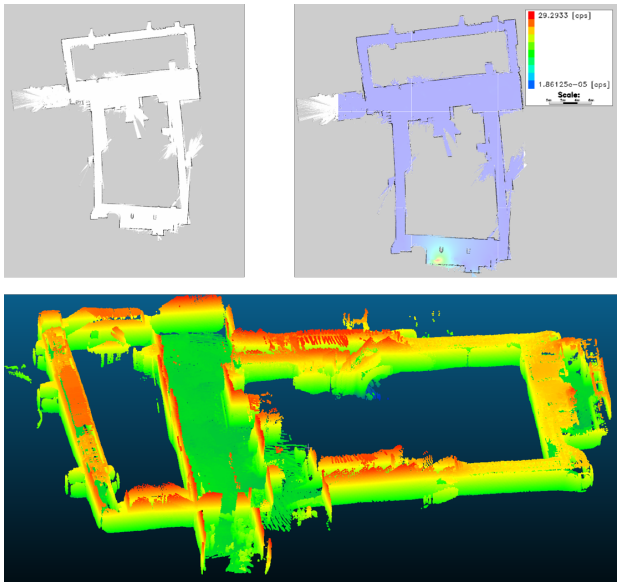


Figure 4. Created maps: upper left) 2D map, upper right) map overlaid with radioactivity measurements bottom) 3D map

map is only partially dense, which means that a dense 3D reconstruction was not possible in all regions. This may be because the range of the 3D depth sensors is too short or because the mobile robot was driven through these regions too fast.

4. Conclusion and Outlook

In this paper the needs of S&R robots with regard to the system requirements were examined from the operator's POV. Furthermore, the approach of the UAS Technikum Vienna to implement the requirements was examined. Search and rescue robots have to cover a wide range of application areas. Starting with the robot's tele-operation, autonomous object recognition and imaging, up to the processing and visualization of sensor data for the operator. The search and rescue robot of the UAS Technikum Vienna is able to generate different maps (2D and 3D), has autonomous capabilities like human victim recognition or autonomous drive and has an easy to use graphical user interface for the operator. The tracker base platform in combination with the robot arm and the end effector allow a high off-road mobility and offer maximum flexibility for manipulation.
Future projects will deal with the tasks of tele-operation of the robot arm with the help of motion controls, the enhancement of the human recognition package by merging the already existing RGB data with point cloud data using Bayesian sensor fusion and visual servoing with reinforcement learning for optimal gripper positioning. Further, to provide real-world S&R capabilities it is necessary to look into possibilities to water-proof the mobile robot.

References

[1] Arctic Robot Challenge. Arctic robot challenge. [Online]. Available:https://arcticrobotchallenge.com/. [Accessed: 2019-07-17].

[2] M. Bjelonic. Yolo ros: Real-time object detection for ros, 2018.

[3] J. Bohren and S. Cousins. The smach high-level executive [ros news]. *IEEE Robotics Automation Magazine*, 17(4):18–20, Dec 2010.

[4] F. Chaumette, S. Hutchinson, and P. Corke. *Visual Servoing*, pages 841–866. Springer International Publishing, Cham, 2016.

[5] B. Choi, G. Park, and Y. Lee. Practical control of a rescue robot while maneuvering on uneven terrain. *Journal of Mechanical Science and Technology*, 32(5):2021, May 2018.

[6] DARPA - Defense Advanced Research Projects Agency. Defense advanced research projects

agency. [Online]. Available: https://www.darpa.mi. [Accessed: 2019-07-17].

[7] D. Doroftei, A. Matos, and G. De Cubber. Designing search and rescue robots towards realistic user requirements. volume 658, 06 2014.

[8] ELROB - The European Land-Robot Trial . The european land-robot trial. [Online]. Available: https://www.elrob.org. [Accessed: 2019-07-17].

[9] ENRICH - The European Robotics Hackathon. European robotics hackathon. [Online]. Available: https://enrich.european-robotics.eu/. [Accessed: 2019-07-17].

[10] EUROC - European Robotics Challenges. European robotics challenges. [Online]. Available: http://www.euroc-project.eu/. [Accessed: 2019-07-17].

[11] D. Fox, W. Burgard, and S. Thrun. The dynamic window approach to collision avoidance. *IEEE Robotics Automation Magazine*, 4(1):23–33, March 1997.

[12] FP7-Icarus. Fp7-icarus. [Online]. Available: http://www.fp7-icarus.eu. [Accessed: 2019-07-17].

[13] S. Grayson. Search & rescue using multi-robot systems. 2014.

[14] A. Hellmund, S. Wirges, Ö. Ş. Taş, C. Bandera, and N. O. Salscheider. Robot operating system: A modular software framework for automated driving. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1564–1570, Nov 2016.

[15] W. Hess, D. Kohler, H. Rapp, and D. Andor. Real-time loop closure in 2D LIDAR SLAM. In *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, pages 1271–1278, May 2016.

[16] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard. OctoMap: An efficient probabilistic 3D mapping framework based on octrees. *Autonomous Robots*, 2013.

[17] Imaging-Development-System-GmbH. Ui-3240le. [Online]. Available: https://de.ids-imaging.com/store/ui-3240le.html. [Accessed: 2020-01-09].

[18] Intel. Depth camera d435 – intel® realsense™ depth and tracking cameras. [Online]. Available: https://www.intelrealsense.com/depth-camera-d435/. [Accessed: 2019-07-17].

[19] H. R. Kam, S.-H. Lee, T. Park, and C.-H. Kim. Rviz: a toolkit for real domain data visualization. *Telecommunication Systems*, 60(2):337–345, Oct 2015.

[20] K. Khoshelham and S. Zlatanova. Sensors for indoor mapping and navigation. *Sensors*, 16(5), 2016.

[21] M. N. Kiyani and M. U. M. Khan. A prototype of search and rescue robot. In *2016 2nd International Conference on Robotics and Artificial Intelligence (ICRAI)*, pages 208–213, Nov 2016.

[22] I. Kostavelis and A. Gasteratos. Robots in crisis management: A survey. In *Information Systems for Crisis Response and Management in Mediterranean Countries*. Springer, Jan. 2017.

[23] G. A. Kumar, A. K. Patil, R. Patil, S. S. Park, and Y. H. Chai. A lidar and imu integrated indoor navigation system for uavs and its application in real-time pipeline classification. *Sensors*, 17(6), 2017.

[24] R. R. Murphy, S. Tadokoro, and A. Kleiner. Disaster robotics. In *Springer Handbook of Robotics*. Springer, Jan. 2016.

[25] J. Oliveira, L. Farçoni, A. Pinto, R. Lang, I. Silva, and R. Romero. A review on locomotion systems for robocup rescue league robots. In H. Akiyama, O. Obst, C. Sammut, and F. Tonidandel, editors, *RoboCup 2017: Robot World Cup XXI*, pages 265–276, Cham, 2018. Springer International Publishing.

[26] S. Pu´tz, J. S. Simón, and J. Hertzberg. Move Base Flex: A highly flexible navigation framework for mobile robots. October 2018.

[27] Rescue Robot League — RoboCup German Open 2019. Rescue robot league. [Online]. Available: https://www.robocupgermanopen.de/de/major/rescue/. [Accessed: 2019-07-17].

[28] C. Roesmann, W. Feiten, T. Woesch, F. Hoffmann, and T. Bertram. Trajectory modification considering dynamic constraints of autonomous robots. In *ROBOTIK 2012; 7th German Conference on Robotics*, pages 1–6, May 2012.

[29] Seibersdorf-Laboratories. Measuring instrument ssm1+. [Online]. Available: https://www.seibersdorf-laboratories.at/en/products/ionizing-radiation/measurement-equipment/measuring-instrument-ssm1. [Accessed: 2020-01-09].

[30] SICK. Tim551. [Online]. Available: https://www.sick.com/at/de/mess-und-detektionsloesungen/2d-lidar-sensoren/tim5xx/tim551-2050001/p/p343045. [Accessed: 2019-07-17].

[31] Taurob GmbH. Ugv-taurob-tracker. [Online]. Available: http://taurob.com/de/produkte-2/ugv-taurob-tracker/. [Accessed: 2019-07-17].

[32] Velodyne LIDAR. Puck-vlp16. [Online]. Available: https://velodynelidar.com/vlp-16.html. [Accessed: 2019-07-17].

Automatic Ontology-based Plan Generation for an Industrial Robotics System

Timon Hoebert, Wilfried Lepuschitz, Munir Merdan
Practical Robotics Institute Austria
{hoebert,lepuschitz,merdan}@pria.at

Abstract. *Programming and re-configuration of robots are associated with high costs, especially for small- and medium-sized enterprises. We present an ontology-driven solution that can automate the configuration as well as the generation of process plans and schedules thereby significantly lowering the efforts in the case of changes. The presented approach is demonstrated in a laboratory environment with an industrial pilot test case.*

1. Introduction

Robotics technology, which can prove high efficiency, precision, and repeatability, is regarded as a viable solution to cope with the increasing number of individualized products. However, robot systems still often do not meet the demands of small- and medium-sized enterprises (SMEs) [8]. Especially, since the programming of industrial robots is complex and time-consuming. To be able to dynamically adapt to new products, robotic systems need to work autonomously. Autonomous systems, in this context, means that robots systems can perform high-level task specifications without explicitly being programmed [2]. To reach specific goals, such systems should be able to receive goals and automatically sequence plans and execute them considering their current state. In our previous work, we presented the control architecture for industrial robots, which can generate actions based on an product model by linking product model, manufacturing process, and production environment in an ontology [7]. In this paper, we focus on the automated plan generation from the ontology and present an approach for flexibly coupling of the decision-making mechanism and ontology.
In section 2, we will detail the architecture and implementation. Finally, Section 3 concludes the paper with a summary and an outlook on further research issues.

2. Architecture

The industrial robot control layer responsible for the management of the robotics systems consists of a World-Model and a Decision-Making component. The decision-making mechanism (Planner) acts as a link between the semantic model of the production environment and the available robot system capabilities. The World Model contains the semantic representation of the relevant objects in the robotics system including their properties and relations. The Planning Domain Definition Language (PDDL) is used for decision-making and the world model is conceptually defined using the Web Ontology Language (OWL) standard. In this context, we transform robotics domain knowledge represented in OWL to PDDL as a targeted mechanism for planning. Multiple applied robotic systems use PDDL for task planning and a lot of work has been done in combining ontologies and AI planning base [5, 1]. Especially ROSPlan [4], a ROS implementation, is a commonly used implementation for this purpose. Based on ROSPlan, OWL-ROSPlan [3] extends this approach using a specialized OWL-Ontology as knowledge-base instead of traditional databases. The disadvantage of these approaches is the implementation effort for application. Even OWL-ROSPlan requires a predefined data format of the ontology. Our work extends this research by automating the translation of the input required by PDDL from the ontology as well as from the PDDL back to the ontology without any predefined ontology formats.

2.1. OWL-PDDL Mapping scheme

The basic building blocks of OWL, are triples consisting of subject, predicate, and object. The basic building blocks of PDDL are actions and PDDL-predicates. To avoid confusion of the two different

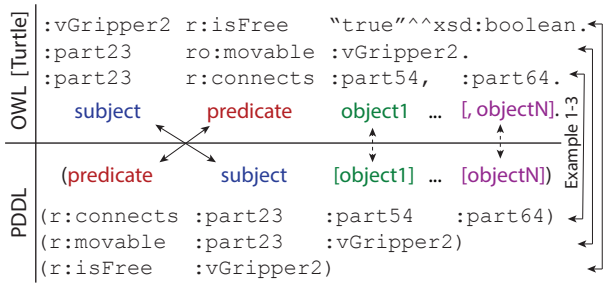


Figure 1. Mapping OWL triples with PDDL predicates. Also, three examples of different parameter length are shown.

types of predicates, the latter ones are only referred to as PDDL-Predicates. The general idea of this approach is the equalization of both building blocks, relating triples with PDDL-predicates. Using a similar approach like WebPDDL [6], OWL-IRIs are used as PDDL-predicate names to identify the data distinctly.

OWL-predicates relate subjects and objects, as verbs do in sentences, but PDDL-predicates are only binary statements relating to multiple object parameters. In practice, PDDL-predicates usually only have one or two object parameters, which can be seen as subject and object. The complete mapping scheme is illustrated with three examples in Figure 1. PDDL-predicates with only one parameter are mapped to boolean-valued objects triples. In practice, PDDL-predicates with more than two parameters are rare because of their complexity (only 4 percent of all predicates from all IPC (1998-2018) domains. But, even these PDDL-predicates can be simplified to multiple PDDL-predicates with two parameters.

2.2. Semantic PDDL Generation

The system automatically generates the PDDL3-problem for the planner based on the information in the ontology and PDDL-domain. This enables easy and extensible programming of the system. The user only has to specify the PDDL-domain with IRIs as PDDL-predicate names and add the goal as triples into a separate part (separate graph) of the ontology database. The system automatically queries all triples of NamedIndividuals regarding this predicates, maps them to PDDL-predicates as mentioned earlier and adds them to the init section in the PDDL-problem. These queries are executed in parallel, and the particular subjects are recorded. After querying the triples, the OWL-types of the recorded subjects are searched in the ontology and written into the PDDL-problem. Since each NamedIndividual can have multiple parent-classes, but not all are relevant for planning, only the ones which are specified in the PDDL-domain are used.

3. Conclusion

The proposed knowledge-driven approach simplifies the programming efforts of the industrial robot. The code for the industrial implementation is generated automatically based on the defined rules, states and actions. A system engineer only needs to describe the functionality of the assembly line or characteristics of the product to be assembled, without having to consider further engineering issues. In our application, we successfully used the developed mechanism for planning pick-and-place operations of an industry robot by Kuka as well as the Festo portal robot, when jointly applied for assembling of PCB boards. As future work, we aim to consider product assembly tasks involving more complex products and production layouts.

References

[1] S. Balakirsky and Z. Kootbally. *An Ontology Based Approach to Action Verification for Agile Manufacturing*, pages 201–217. Springer International Publishing, Cham, 2014.

[2] G. A. Bekey. *Autonomous Robots: From Biological Inspiration to Implementation and Control (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.

[3] L. Buoncompagni, A. Capitanelli, and F. Mastrogiovanni. A ros multi-ontology references services: Owl reasoners and application prototyping issues. *arXiv preprint arXiv:1706.10151*, 2017.

[4] M. Cashmore, M. Fox, D. Long, D. Magazzeni, B. Ridder, A. Carrera, N. Palomeras, N. Hurtos, and M. Carreras. Rosplan: Planning in the robot operating system. In *Twenty-Fifth International Conference on Automated Planning and Scheduling*, 2015.

[5] M. Crosby, R. P. A. Petrick, F. Rovida, and V. Krüger. Integrating mission and task planning in an industrial robotics framework. In *ICAPS*, 2017.

[6] D. Dou. The formal syntax and semantics of webpddl. Technical report, Technical Report, Technical report, University of Oregon, 2008.

[7] T. Hoebert, W. Lepuschitz, E. List, and M. Merdan. Cloud-based digital twin for industrial robotics. pages 105–116, Cham, 2019. Springer International Publishing.

[8] A. Perzylo, N. Somani, S. Profanter, I. Kessler, M. Rickert, and A. Knoll. Intuitive instruction of industrial robots: Semantic process descriptions for small lot production. 10 2016.

How does explicit exploration influence Deep Reinforcement Learning?

Jakob J. Hollenstein, Erwan Renaudo, Matteo Saveriano, Justus Piater
University of Innsbruck
{jakob.hollenstein,erwan.renaudo,matteo.saveriano,justus.piater}@uibk.ac.at

Abstract. *Most Deep Reinforcement Learning (D-RL) methods perform local search and therefore are prone to get stuck in non-optimal solutions. To overcome this issue, we exploit simulation models and kinodynamic planners as exploration mechanism in a model-based reinforcement learning method. We show that, even on a simple toy domain, D-RL methods are not immune to local optima and require additional exploration mechanisms. In contrast, our planning-based exploration exhibits a better state space coverage which turns into better policies than the ones learned via standard D-RL methods.*

1. Introduction

Deep-Reinforcement Learning (D-RL) has shown promising results in challenging robotics domains (*e.g.* [4]), but can be resource demanding and difficult to train. We assume that part of the difficulty of learning good policies is related to insufficient exploration. Other D-RL methods like [1, 3, 6] partially address the problem by increasing the number of training steps, or by relying on the environment implementation to provide *exploring-starts* to cover a diverse enough state-space region. However, these solutions are impractical and potentially dangerous in robotics applications.

In the robotic context, directed exploration via physically-based simulation appears more promising to find good solutions more reliably and in less time. Therefore, this work proposes the *Planning for Policy Search (PPS)* method that exploits a kinodynamic planner in the exploration phase to collect data which are then used to learn a policy, thereby eliminating the planning time during execution. PPS is tested on the point mass system described in Table 1 and compared with D-RL approaches.

This research has received funding from the European Union's Horizon 2020 research and innovation programme (grant agreement no. 731761, IMAGINE)

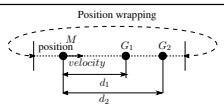
Dynamics
$X = \begin{bmatrix} x \\ \dot{x} \end{bmatrix}$ $A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ 
$\dot{x} = Ax + Bu$
Reward
$\max((1 - \tanh \lvert X - G_1^* \rvert), 2(1 - \tanh \lvert X - G_2^* \rvert))$ $G_1 = \begin{bmatrix} -2.5 \\ 0.0 \end{bmatrix}$ $G_2 = \begin{bmatrix} 6.0 \\ 0.0 \end{bmatrix}$
Limits
$u \in [-1; 1]$ $x \in [-10; 10]$ $\dot{x} \in [-2.5; 2.5]$

Table 1. Description of the 1D double-integrator test environment: a point mass M can be moved in a one-dimensional space position-velocity $X = [x, \dot{x}]$ by applying a continuous-valued force. Reward is received based on the distance to two possible goal locations (G_1, G_2).

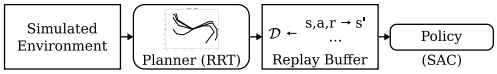


Figure 1. Illustration of PPS Method

2. Planning for Policy Search

The presented PPS implementation (Figure 1) consists of a Linear Quadratic Regulator (LQR)-Rapidly Exploring Random Tree (RRT) [5] to create a tree of data $\mathcal{D} = \{(s, a, r, s'), \ldots\}$ from which Soft-Actor Critic (SAC) [1] learns a policy. In contrast to [5] quadratic programming-based finite-horizon steering is used to extend the tree. In our setup, all the environment interaction data created by RRT are used as training data for the policy rather than using only successful trajectories as expert demonstrations.

3. Evaluation

PPS is evaluated in the one-dimensional goal reaching task presented in Table 1. The environment contains two distinct goal locations. The agent receives a reward based on the distance to the goal

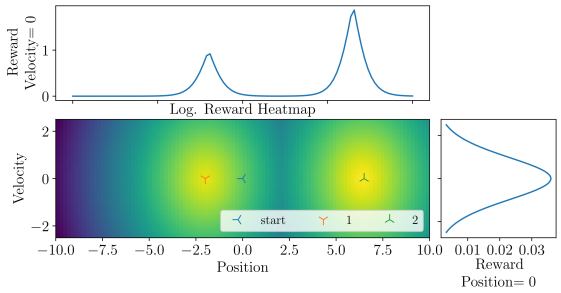


Figure 2. The reward (heatmap) and reward distributions (plots above and on the right of the heatmap) for the double-integrator. The agent starts at $x = 0$ with $\dot{x} = 0$. The reward is based on the distance of the agent to the goal positions 1 and 2.

Alg.	DDPG	PPO	SAC	**PPS (RRT)**
Non-Ex.	15.5%	20.8%	20.4%	**79.3%**
Ex.	59.0%	60.9%	61.0%	-

Table 2. Final coverage as percent of visited bins.

points. The goals ($x_1 = -2.5$ and $x_2 = 6.0$) are chosen such that simply maximizing the reward from the starting position leads to a suboptimal policy, i.e. a local optimum (see Figure 2).

We compare the performance of PPS against the prominent D-RL algorithms Proximal Policy Gradient (PPO) [6], Deep Deterministic Policy Gradient (DDPG) [3], and SAC [1], using the implementation in [2].The algorithms are run for 10^5 environment steps; the D-RL algorithms use 100-step episodes. To have a broader baseline we included an *exploring-starts* mechanism where the initial state of the double integrator is sampled uniformly. However, especially in robotic tasks, exploring starts are impractical and potentially dangerous and should be avoided.

We first compare the state-space coverage obtained from data collected during the exploration phase of the different D-RL approaches. The coverage is calculated as the percentage of non-empty, uniformly-shaped bins. The number of bins is set to $\sqrt{10^5/5}$ in each dimension, i.e. we expect 5 data points in each bin on average. See Table 2 for the final coverages.

Second, Figure 3 depicts boxplots of the evaluation returns achieved by the D-RL algorithms after training for 10^5 steps. DDPG achieves higher rewards without exploring starts, while PPO and SAC profits from exploring starts. Our PPS method shows improved performance compared to non-exploring starts methods. Moreover, the policies learned with PPS achieve performance comparable to the directly-trained SAC policy with exploring starts.

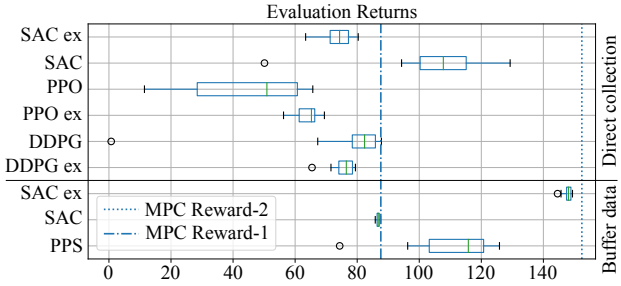


Figure 3. Box plot of the return distributions (11 independent runs); each run consists of the mean of 10 evaluation runs. The evaluation runs are performed towards the end of the training process, equally spaced 10 learning episodes apart.

4. Discussion

In this work, we highlighted that standard D-RL algorithms are not immune to getting stuck in suboptimal policies even in a toy problem with two local optima. The agent controlled by PPS explores a wider part of the state space than D-RL methods that focus on reward accumulation, even with exploring starts. The data gathered by RRT are not biased by reward accumulation and is thus more representative of the environment.

References

[1] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. In *Int. Conf. Machine Learning (ICML)*, 2018.

[2] A. Hill, A. Raffin, M. Ernestus, A. Gleave, R. Traore, P. Dhariwal, C. Hesse, O. Klimov, A. Nichol, M. Plappert, A. Radford, J. Schulman, S. Sidor, and Y. Wu. Stable Baselines. *GitHub repository*, 2018.

[3] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra. Continuous control with deep reinforcement learning. In *Proc. 4th Int. Conf. Learning Representations, (ICLR)*, 2016.

[4] OpenAI, M. Andrychowicz, B. Baker, M. Chociej, R. Józefowicz, B. McGrew, J. W. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, J. Schneider, S. Sidor, J. Tobin, P. Welinder, L. Weng, and W. Zaremba. Learning dexterous in-hand manipulation. *I. J. Robotics Res.*, 39(1), 2020.

[5] A. Perez, R. Platt, G. Konidaris, L. Kaelbling, and T. Lozano-Perez. LQR-RRT*: Optimal sampling-based motion planning with automatically derived extension heuristics. In *IEEE Int. Conf. Robotics and Automation*, 2012.

[6] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal Policy Optimization Algorithms. *CoRR*, abs/1707.06347, 2017.

UGV Radiation Mapping using a Particle Filter

Alexander Permann, Daniel Hettegger, Gerald Steinbauer
Institute of Software Technology, TU Graz
alexander.permann@student.tugraz.at, daniel.hettegger@alumni.tugraz.at
steinbauer@ist.tugraz.at

Abstract. *We present and evaluate a particle filter based approach to predict the location and emission intensity of an arbitrary and unknown number of stationary nuclear radiation sources from measurement data taken by an autonomously navigating unmanned ground vehicle (UGV).*

1. Introduction

Due to the threat for humans caused by radiation and the associated difficulties after a nuclear disaster it is crucial to establish save methods of estimating the radiation distribution in certain affected areas. For this purpose we suggest to record radiation measurement using an autonomous UGV. These measurements are then processed by an adapted particle filter to generate a detailed radiation distribution model of the affected area. The approach presented in this paper has been successfully tested in realistic conditions at the *ENRICH 2019 — European Robotics Hackathon*, where live radiation sources had to be detected inside the nuclear power plant in Zwentendorf, Austria.

2. Related Research

In [1] Eric T. Brewer used an autonomously flying aerial platform to detect and locate a single radioactive point source using a particle filter. In [2] M. Morelande et al. compare the performances of a maximum likelihood estimator and a Bayesian estimator approach to deal with an unknown number of sources. D. Shah et al. present a particle filter in [3] that manages to locate multiple radiation sources.

3. Problem Description

The setting is represented by a set Θ of unknown radiation sources s and a set Γ of radiation measurements m. The goal is to generate a set Ψ of estimated sources $\hat{s}$, that fits the number and intensities of the real sources accurately. Each set holds elements defined by a certain location x_i and y_i and an equivalent radiation dose rate α_i in $\mathrm{Sv\,s^{-1}}$ that either represents the actual measurement for the set Γ or the theoretical dose rate that would be measured at the exact position of a source for the sets Θ and Ψ. In general for modelling the radiation intensity at a certain location l based on a set of sources Θ, we assume that the radiation follows the principle of *superposition* and the *inverse-square-law* which has been shown to be applicable by multiple former approaches. [1, 2]:

$$\alpha(l) = \alpha_{bgr} + \sum_{s \in \Theta} \frac{\alpha_s}{4 \cdot \pi \cdot d_s(l)^2} \qquad (1)$$

where α_{bgr} denotes the known background radiation and $d_s(l)$ the euclidean distance between the location l and the source s.

4. Particle Filter

In contrast to common particle filter use cases in robotics (e.g., estimating a robots position), it is now necessary to detect multiple sources that can co-exist at the same time at different positions. In this context particles are predictions of potential sources [3] with each particle $p \in P$ being represented similar to real sources by $p =< x_p, y_p, \alpha_p, w_p >$ with an additional weight w_p that is related to the probability that a certain particle has the parameters of a real source. At first the particles are initialized uniformly distributed on the plane where the measurements took place and given a random intensity within the same range of the measurement results. The algorithm then iteratively performs the two steps of *weighting* and *re-sampling* and adds estimated sources $\hat{s}$ to the growing set Ψ until a maximum number of iterations T is reached and Ψ represents a consistent estimation for Θ based on the measurements Γ.

4.1. Weighting

First an intensity estimation $\hat{\alpha}_m$ for a certain measurement m is calculated based on the single particle $p \in P$ to be weighted and the influence of all already defined sources $\hat{s} \in \Psi$ assuming the model presented in Equation 1:

$$\hat{\alpha}(m) = \alpha_{bgr} + \frac{\alpha_p}{4 \cdot \pi \cdot d_p(m)^2} + \sum_{\hat{s} \in \Psi} \frac{\alpha_s}{4 \cdot \pi \cdot d_s(m)^2} \tag{2}$$

Using Equation 2 the relative mean square error considering all measurements is calculated like:

$$e_{rmse} = \frac{1}{|\Gamma|} \sum_{m \in \Gamma} \left(\frac{\hat{\alpha}_m - \alpha_m}{\hat{\alpha}_m} \right)^2 \tag{3}$$

where $|\Gamma|$ is the number of measurements. The weight for a single particle is then calculated like:

$$w_p = \frac{1}{1 + e_{rmse}} \tag{4}$$

After all particle weights have been updated the weights are normalized such that $\sum_{p \in P} w_p = 1$.

4.2. Re-Sampling and Clustering

During re-sampling a certain percentage of particles with the highest weights stay the same, while another percentage of particles with the smallest weights are omitted and newly drawn from a uniform distribution over the search space. The remaining particles are re-sampled by adding Gaussian Noise to the intensity α_p and position based on the particles weight:

$$[x'_p, y'_p, \alpha'_p]^T \sim \mathcal{N}\left([x_p, y_p, \alpha_p]^T, \frac{\text{diag}(\sigma_{pos}, \sigma_{pos}, \sigma_{int})}{1 + w_p} \right) \tag{5}$$

The total number of particles stays the same. After a defined number of iterations k the particles are clustered using the mean shift algorithm as suggested by [3]. The cluster centroids have the same structure as a single particle and are then evaluated by the weighting algorithm described in section 4.1. If the weight of a cluster surpasses a defined threshold φ the centroid is believed to be a real source and added to the growing set of predicted sources Ψ.

5. Experimental Evaluation - ENRICH 2019

As part of the TU Graz Robotics Team TEDUSAR we participated in the European Robotics Hackathon - ENRICH 2019 at the nuclear power plant Zwentendorf, Austria[1] and were able to test our particle filter approach under real world conditions. An autonomous robot created a 3D map of the interior while our approach created the mathematical model of the real radiation sources and the radiation contamination. The parameters used are shown in Table 1.

Total Particles	2000
Random New Particles	10 %
Sustain Particles	10 %
Max Iterations T	1000
Confidence Threshold φ	0.9
Clustering Interval k	20 iter
Position Deviation σ_{pos}	0.5
Intensity Deviation σ_{int}	0.4

Table 1: Hyper-parameters used for the ENRICH 2019

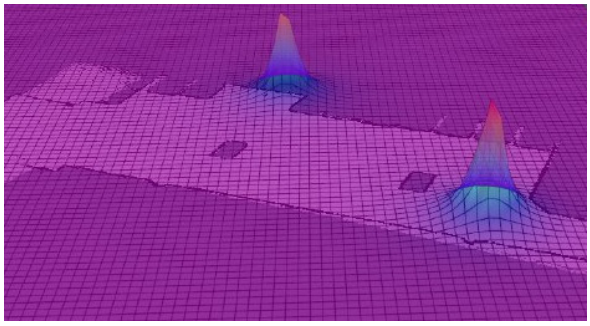


Figure 1: Source estimation based on live measurement data during the ENRICH 2019 in Zwentendorf.

An experimental result can be observed in Figure 1. In this experiment two sources were placed in a larger room. After traversing the room and collecting radiation measurements our approach correctly predicted the location and intensity of the two sources.

6. Conclusion and Future Work

In this paper we presented the adaptation of an approach based on a particle filter to determine the location and intensity for an arbitrary and unknown number of stationary radiation sources. This approach has been successfully tested and proven to be applicable in real world scenarios, like an accident in a nuclear facility. Future work will focus on reducing the number of hyper-parameters.

[1] www.enrich.european-robotics.eu

References

[1] E. T. Brewer. Autonomous localization of $1/r^2$ sources using an aerial platform. Master's thesis, Faculty of the Virginia Polytechnic Institute and State University, Blacksburg, Virginia, 2002.

[2] M. Morelande, B. Ristic, and A. Gunatilaka. Detection and parameter estimation of multiple radioactive sources. In *2007 10th International Conference on Information Fusion*, pages 1–7, July 2007.

[3] D. Shah and S. Scherer. Robust localization of an arbitrary distribution of radioactive sources for aerial inspection. *CoRR*, abs/1710.01701, 2017.

Towards ASP-based Scheduling for Industrial Transport Vehicles

Felicitas Fabricius
Marco De Bortoli
Gerald Steinbauer
Graz University of Technology
{mbortoli,steinbauer}@ist.tugraz.at

Maximilian Selmair
BMW Group
maximilian.selmair@bmw.de

Michael Reip
Incubed IT
m.reip@incubedit.com

Martin Gebser
Universität Klagenfurt
martin.gebser@aau.at

Abstract.

The increasing number of robots and autonomous vehicles involved in logistics applications leads to new challenges to face for the community of Artificial Intelligence. Web-shop giants, like Amazon or Alibaba for instance, brought this problem to a new level, with huge warehouses and a huge number of orders to deliver with strict deadlines. Coordinating and scheduling such high quantity of tasks over a fleet of autonomous robots is a really complex problem: neither simple imperative greedy algorithms, which compromises over the quality of the solution, nor precise enumeration techniques, which make compromises over the solving time, are anymore feasible to tackle such problems. In this work, we use Answer Set Programming to tackle real-world logistics problems, involving both dynamic task assignment and planning, at the BMW Group and Incubed IT. Different strategies are tried, and compared to the original imperative approach.

1. Introduction

Industry 4.0 is bringing more and more interest toward the digitalization of all productive stages in the industrial field. Even before that, we all have been witnesses of the big impact robotics had in industry, by the automatization of repetitive tasks. In the last years, thanks to the increasing computational power, Artificial Intelligence (AI) is spreading as well, leading to the next step of the integration between robots and production: the automatization of complex tasks requiring reasoning. In this perspective, optimization of logistics is crucial for large companies, in order to save both time and money. Still we are dealing with a production environment which considers a fleet of robots floating around, efficiently performing tasks and carrying goods where to model such NP-hard domains a high number of constraints is needed. For this reason, a imperative approach become more and more difficult to maintain, and cannot benefit from the numerous meta-heuristics and optimizations (if not manually implemented) already encoded inside the solvers of other programming paradigms, like declarative programming. Answer Set Programming (ASP) is a fast and intuitive logic language, which already has many applications both in industry and in research (see Section 2). In this paper we are going to investigate the difference between the two paradigms, by replacing the imperative part of the task schedulers used by two companies, the BMW Group and Incubed IT, with an ASP implementation. While classical languages are well suited for greedy algorithms implementation, declarative programming has other advantages: first of all, the focus is on the description of the problem, leaving all the solving details to the external solver. Moreover, most solvers are configurable with a lot of meta-heuristics to cut the search space: the user has only to find the one which fits the problem better, without implementing anything. Then, since logic languages are basically based on enumeration techniques, an ASP solver looks for the best solution, or at least the best one in a given time. Depending on the size of the instance, this behaviour leads to huge computational time with respect to a greedy al-

gorithm. We are interested into the analysis of this trade-off between greedy solving time and declarative solution quality. This paper is based on Felicitas Fabricius' Master thesis [6].

2. Related Research and ASP Foundations

The demand for increased complexity and scalability in industry automatization requires more and more powerful techniques and algorithms. Imperative programming is suitable to write a very problem-specific solution. However, the development of such kind of code can be really arduous, time expensive and difficult to maintain. Optimal task scheduling and planning, enriched by domain-based heuristics, requires a huge amount of code lines if written with an imperative language [17].

Answer Set Programming, and logic programming in general, allows to tackle combinatorial problem in a very intuitive way, splitting the work into two phases: the description of the problem and its efficient solving procedure [7]. The programmer has only to care of the former, and this requires just a fragment of the effort required by an imperative language. Then, (s)he can use one of the solvers available in the market, like Clingo or DLV, to find the optimal solution, improving it with a large set of meta-heuristics. Although the most common approach is the imperative one, many use-cases of ASP applied to industry can be found in the literature: in 2017, Dodaro and Maratea designed a shift plan for 164 Italian nurses, calculating the optimal plan for an entire year in just 50 minutes, using the state-of-the-art solver Clingo [3]. Staying in the shift scheduling field, the DLV solver was deployed to find the optimal shift plan for seaport workers [14]. In this case, the problem was complicated by the fact that the employees have different qualifications, and there were different kinds of tasks. Finding of the optimal one month-long plan required 8 minutes. A similar work, considering different demands as well, is described in [2]. Moving to other kind of industrial fields, ASP was used in E-tourism in order to find the travel which suits the user the most [1]. In [12] the authors used ASP for phone routing in call centers. The customer was classified in a category and assigned directly to the human operator. We can find plenty of ASP applications regarding task assignment and routing as well. Examples can be found in [4], [16], [10], [13] and [15].

Answer Set Programming is based on the stable model semantics, presented by Gelfond and Lifschitz in [11] for dealing with logic programs with negation as failure. With the following we give a quick overview of the language semantics [2, 7].

A *rule* r in a logic program is an expression of the form

$$h \leftarrow a_1, \ldots, a_m, \neg a_{m+1}, \ldots, \neg a_n \qquad (1)$$

where $a_1, \ldots, a_n$ are *atoms* of the form $s(t_1, \ldots, t_k)$, in which s is a *predicate* symbol and $t_1, \ldots, t_k$ are *terms*, viz. constants, variables, or functions, and $\neg$ stands for *default negation*. The head h of r is either an atom a, a choice $\{a\}$, or the special symbol $\bot$. If h is an atom and $n = 0$, we call r a *fact*, a *choice rule* if h is $\{a\}$, and an *integrity constraint* if h is $\bot$; we skip $\leftarrow$ or $\bot$, respectively, when writing rules (1) with $n = 0$ and integrity constraints. A *logic program* P is a set of rules and constraints. In the first-order case, terms occurring in P may include arithmetic expressions, and atoms may be based on relational operators like "$<$". On the other hand, a term, atom, rule, constraint, or program is *ground* if it does not include variables, arithmetic expressions, or relational operators. A first-order program P stands for the set $grd(P)$ of all instances of rules and constraints constructible by substituting ground terms for variables and evaluating arithmetic expressions as well as relational operators in the standard way. For details on ground instantiation, we refer the interested reader to [5, 9]. The semantics of a logic program P is given by its stable models, which are particular sets of (true) ground atoms as defined in the following. The *reduct* P^X relative to a set X of ground atoms is the set of all rules and constraints in $grd(P)$ such that $\{a_1, \ldots, a_m\} \subseteq X$, $\{a_{m+1}, \ldots, a_n\} \cap X = \emptyset$, and $a \in X$ if $h = \{a\}$ is a choice for a rule (1). Then, X is a *stable model* of P if it is $\subseteq$-minimal among the sets of ground atoms such that, for all rules in P^X, $\{a_1, \ldots, a_m\} \subseteq X$ implies $h \in X$ or $a \in X$ if $h = \{a\}$. In addition to rules, a logic program can contain `#minimize` statements of the form

$$\texttt{\#minimize}[\ell_1 = w_1@L_1, \ldots, \ell_n = w_n@L_n].$$

Besides literals ℓ_i and integer weights w_i for $1 \leq i \leq n$, a `#minimize` statement includes integers L_i providing priority levels [8]. The `#minimize` statements in P distinguish optimal answer sets of P in the following way. For any set X of atoms and integer L, let Σ_L^X denote the sum of weights w_i such that

$\ell_i = w_i@L$ occurs in some `#minimize` statement in P and ℓ_i holds w.r.t. X. We also call Σ_L^X the utility of X at priority level L. An answer set X of P is dominated if there is an answer set Y of P such that $\Sigma_L^Y < \Sigma_L^X$ and $\Sigma_{L'}^Y = \Sigma_{L'}^X$ for all $L' > L$, and optimal otherwise.

3. ASP and Logistics: Two Cases-Studies

To evaluate ASP in an industrial environment, we discovered two interesting case-studies. Both are related to Fleet Management Systems (FMS) - one at Incubed IT, the other one at the BMW Group. In both cases, the imperatively described task allocation strategy was replaced by an ASP-based program.

3.1. The BMW Use Case: Task Assignment and Charging Management

By the following, the requirements for the FMS at the BMW Group are described. Here, two elemental decisions have to be made. These are on one hand the assignment of tasks to the vehicles and on the other hand the assignment of charging and parking stations to the same vehicles. Both decisions are made online, which means that neither tasks nor the needs for charging (and parking) are known beforehand. With *task* we mean a transportation job of a container, accomplished by a vehicle, from a station to another one. The required time is estimated from the Euclidean distance.

For the task assignment, the standard C# scheduler applies a trivial first-in-first-out (FIFO) strategy, which means that earlier created tasks have to be executed first. By that, the criterion for the selection of tasks, formulated as a constraint, is not to assign a task if there is another appropriate task with earlier creation time assignable. Vehicles on the field must have a battery level at a minimum of 25 %, and charging vehicles a battery level of 40 % to be assigned to tasks. The optimal assignment of vehicles to tasks is based on the traveling costs that are set to be the Euclidean distance between robots and the first goal of the assigned task. The used optimization criterion ensures the lowest traveling cost for the tasks with earliest time of creation.

In ASP a different optimization criterion is used, in order to achieve a better overall quality of the solution. The Euclidean distance for all assignments is summed up and minimized, in order to have a better make-span and save more energy. Considering T and R as the set of tasks and robots respectively, task assignment is encoded by the following logic formulas:

$$\forall t \in T(|\{r \in R|(assign(t, _)\}| \leq 1)$$
$$\forall t_1, t_2 \in T, \forall r \in R$$
$$(assign(t_1, r) \wedge assign(t_2, r) \wedge t_1 \neq t_2 \Rightarrow \bot)$$

The first formula may (non-deterministically) assign each task to one robot at most. The second one makes sure that two different tasks are not assigned to the same robot. The non-deterministic choice is driven by the optimization algorithm. In ASP, above formulas are encoded as follows (:- stands for $\leftarrow$):

Listing 1 ASP encoding of the task assignment

```
0{assign(T,R):robot(R,_,_,_)}1:-
  task(T,_,_).
:- assign(T,R), assign(T2,R), T != T2.
```

The first rule makes use of both a conditional literal and a cardinality constraint. A conditional literal $a : b_1, \ldots, b_n$ is a nested implication, where a and $b_1, \ldots, b_n$ can be seen as the head and the body of a rule respectively. The cardinality constraint is used to ensure that each task is assigned to one robot at most. Given $x\{head\}y$:- $body$, the meaning is that, for each different $body$ instantiation (for each task T in our case), the $head$ is instantiated from x to y times (from 0 to 1 in our case). In our code this implies that, for each task T, at most one robot R is assigned inside the head. The second rule is an integrity constraint. In the case that after the task assignment was performed unassigned vehicles are remaining, these free vehicles are assigned to charging stations and parking places. The rules used for this particular assignment problem are defined separately for vehicles on the field and vehicles currently in charging stations. A charging vehicle can only be assigned to a charging station if the battery level is below 90 %. Vehicles on the field can be sent to charging stations any time, regardless of the current battery level. Charging vehicles can go to a parking place only if the battery level is above or equal 90 %, whereas vehicles on the field can go to parking places independently from the battery level. In the original implementation, priority is given to vehicles with the lowest battery level. Similarly to the FIFO strategy in task assignment, first we assigne the least charged vehicle to the closest station, then the second least charged one, and so on. However, this implementation shows its limits on circumstances where multiple robots have critical battery levels that differ

only in a very small amount. For this reason, and since our goal with the declarative encoding is to improve the overall quality of the assignment, in the ASP implementation we minimize the overall travelling distance, like we do for the task assignment. The rules and constraints needed are very similar to the ones we used before, where PR1, PR2 are some user-defined parameters required for the assignment:

Listing 2 ASP encoding of the charge assignment - assignment rule

```
0{charge(S,R): station(S,PR1,PR2)}1
        :- robot(R,PR1,PR2,_).
```

3.2. The Incubed IT Use Case

Incubed IT is a robotics company focused on software development for smart robots. They typically deal with problems of the same type as the previous use-case we just discussed above. Thus the main topic is multi-robot planning and scheduling. For this reason, programmers at Incubed IT designed a highly parameterized platform which, if configured accordingly, can face a lot of different situations, like warehouses of online traders, logistic centers of supermarkets and car manufacturing plants. Fortunately, this platform is quite modular, partially centralized and partially decentralized, with a main FMS module which is responsible for the coordination of the many parts of the system. Thanks to this design, replacing the old solving module with the ASP solver has been easy to do.

In the imperative implementation, two kinds of optimization costs can be used: FIFO and global optimum. The former does not require more explanations, while the latter considers a priority number associated to each task. Regarding the task assignment, we stick to the important constraint rule in ASP: we can assign only one vehicle to a task, and only one task to a vehicle at a time. The same rules and constraints we used for the BMW use-case thus fit to Incubed IT software as well.

We can now focus on the other problem to solve, the charge assignment. The charging strategy here is more sophisticated than in the BMW-case, and robot can be sent to charge for four different reasons: *fixed time slot charging*: robots are assigned to charging stations due to a reached time slot; *critical charging*: robots are assigned to charging stations due to a battery level below the critical charging limit; *busy charging*: robots are assigned to charging stations due to a battery level below the busy charge limit; *idle charging*: robots are assigned to a charging station due to not enough appropriate assignable tasks. Obviously, all of these parameters (critical and busy charging limit, duration of the time slot) can be customized by the user. We define now the rules and constraints used to implement the third situation:

Listing 3 Assignment of busy charging robots

```
0{charge(S,R_busy) : chargingstation(S
    ,_,_,_,_), robot_station(R,S)}1 :-
robot_charge_opt(R,BL,automatic_mode,_
    ,_,_,BCL,CCL), BL <= BCL, BL > CCL.
```

Listing 4 Avoidance of double allocations

```
:- charge(S,R,_),charge(S,R2,_),R!=R2.
:- charge(S,R,_),charge(S2,R,_),S!=S2.
:- assign(T,R),charge(_,R,_).
```

In contrast to the BMW case-study, here we do not handle the two problems of task and charge assignment separately: we optimize two different weighted criteria. The most important one is the minimization of the overall travelling distance of robots assigned to tasks or to charging stations due to forced time-slot, critical or busy charging. Then, the same optimization, with a lower weight, is applied to robots assigned to parking places and charging stations for idle charging.

4. Evaluation of Runtime and Quality for both Case-Studies

In this section, we present a brief evaluation of both case-studies. We designed several instances for each case-study involving different numbers of robots, orders, charging and parking stations to test different scales. Subsequent, the runtime as well as the quality of the solutions for these scenarios are compared. Furthermore, since Clingo can combine different meta-heuristics and parallelization strategies for the solving process, we tested all the combinations between them in order to find, for each case, the best one. As a result of the evaluation of these solving approaches [6], we chose the branch-and-bound-based optimization strategy in combination with splitting-based search multithreading and four threads for the two BMW assignment problems, while for Incubed IT the best approach is the Vsids Heuristic combined with compete-based multithreading with four threads.

The systems of BMW and Incubed IT have been tested on devices with the following specifications. At BMW an Intel(R) Core(TM) i5 with a 1.70GHz processor and 8GB RAM is used. At Incubed IT an Intel(R) Core(TM) i5- 7200U is used with a 2.50GHz processor and 8GB RAM. On both systems Windows 10 is installed. Clingo is running in version 5.3.0 with Gringo V5.3.0. and Clasp V3.3.4.

4.1. Evaluation at BMW Group

In Table 1, the mean value and the standard deviation of the runtimes for all test scenarios (10 for each scenario) are shown and the number of solved test runs is given. If the optimal solution is not found within the BMW-specific time limit of 60 seconds, the solving process is aborted. Consequently, these aborted test runs are not considered in the calculations for the mean and standard deviation. The mean performance of the imperative method is for every scenario the best. As shown in the tables, two different ways of using ASP were tested. In the first one, the solver is directly called inside C#, while in the second we run ASP standalone. The serious performance issues of the former indicate potential for an improved incorporation of the ASP call in C#.

The instances are formed as follow: for the test scenario 1, we have 5 tasks and 5 robots; for the scenario 2, 20 tasks and 12 robots; finally, scenario 3 has 50 tasks and 30 robots. The positions of the robots and stations of the tasks are randomly placed on a $1000\,\mathrm{m} \times 1000\,\mathrm{m}$ area.

Looking at the results in Table 1 the imperative solution seems the winner, but in ASP not the Euclidean distance for single robot is optimised, but the traveling costs of the whole fleet. So, by using ASP, we are rewarded with far better quality solutions, as witnessed by Table 4, where traveling costs for scenario 3 are shown. This scenario is particularly interesting, since ASP was not able to find the provable optimal solutions within the time limit. Although, while looking for that, solvers like Clingo keep returning the best solution found so far, as soon as it finds a better one. Looking at Table 4, we can see that the best ASP solution found within 1 second considerably beats the C# solution. However, in this scenario we do not get an improvement with higher time limits. Results with the other scenarios are similar, with the imperative implementation never being close to the ASP traveling distance. This particular problem highlights the performance-quality trade-off between the two approaches.

In Table 2 the mean value and the standard deviation of the runtime of every test scenario is shown, considering the charge and park problem. Same rules as before are applied regarding the time limit of 60 seconds. The instances are formed as follow: 2 charging stations (CS), 3 parking places (PP) and 3 robots (R) for scenario 1; 7 CS, 14 PP and 3 R for scenario 2; finally, 17 CS, 33 PP and 30 R for scenario 3.

The imperative C# approach shows for all scenarios a better performance than the ASP-implementations, which, as in the task assignment problem, makes use of a different optimization, minimizing the overall travelling distance between robots and stations, while the C# program prioritizes the robots with the most critical battery level. In contrast to the task assignment, in this case the problem is too complex to ASP, which does not succeed in finding good quality solutions (Table 4) and, in some cases, it does not succeed to find a solution at all. This observation leads to the assumption that the encoding of the park and charge assignment problem in ASP is not optimal, as the performance of the task assignment encoding for similarly scaled instances is significantly better.

4.2. Evaluation at Incubed IT

In the Incubed IT use-case, the two problems, task assignment and park and charge assignment, are handled together, according to our characterization in the previous section. In Table 3, the mean value and standard deviation of the runtimes for the test scenarios solved with the original code and with the in-Java integrated ASP are shown, together with standalone ASP. A timeout is reached when a test run requires more than 30 seconds to find an optimal solution. Test runs that reached the timeout are not considered in the calculation for the mean and the standard deviation. The testing environment has a floor area of $100\,\mathrm{m} \times 86\,\mathrm{m}$ where the robots are freely movable. The three scenarios we are going to test are formed as follows: for scenario 1, we have 5 robots (R), 3 charging stations (CS), 7 parking places (PP) and 5 tasks (T); 10 R, 6 CS, 14 PP and 10 T for scenario 2; finally, for the last scenario we have 30 R, 18 CS, 42 PP and 15 T.

As we would expect from an NP problem solver, the reader can notice from the results that ASP is faster than the Java program while solving small

Test Scenario	C# Implementation			ASP within C# Implementation			Standalone ASP		
	μ [ms]	σ [ms]	# TRS	μ [ms]	σ [ms]	# TRS	μ [ms]	σ [ms]	# TRS
1	0.00	0.00	10	415.50	18.16	10	8.30	8.92	10
2	0.30	0.48	10	2,802.20	4,445.21	10	1,428.90	2,746.91	10
3	0.00	0.00	10	/	/	0	/	/	0

Table 1: Runtime and solved test runs (TRS) for the different BMW task assignment implementations

Test Scenario	C# Implementation			ASP within C# Implementation			Standalone ASP		
	μ [ms]	σ [ms]	# TRS	μ [ms]	σ [ms]	# TRS	μ [ms]	σ [ms]	# TRS
1	0.00	0.00	10	473.80	81.24	10	13.90	8.88	10
2	16.20	4.87	10	788.10	350.75	10	341.70	404.17	10
3	1,753.30	127.58	10	/	/	0	/	/	0

Table 2: Runtime and solved test runs (TRS) for the different BMW park and charge assignment implementations

instances, regardless of the number of constraints. Though, once the size of the problem hits the combinatorial blow-up point, it fails to return an optimal plan within time.

To measure the quality of the solution, we consider the two metrics we described in the previous section: most important are the overall travelling costs for task assignment and critical charging; the travelling distance of the other kinds of assignment (like for parking places) are then considered. Since the optimization strategy adopted with ASP is very similar to the one already used in the original program, in all the scenarios in which the optimal declarative solution is found within time, its quality w.r.t. to these metrics coincides to the Java solution quality. For this reason, like we did for the BMW case, when the ASP solver fails to find the optimal solution within the limit, we are interested in the analysis of the best ASP solution found so far. This situation shows up in the third scenario. Looking at Table 4, where the total cost (which is the weighted sum of the two metrics) is shown, it can be seen that the original implementation in Java provides a significantly faster and better solution than the ASP implementations.

5. Conclusion

The goal of this work is to make a comparison, in different real-world logistics scenarios, between the classic imperative paradigms and the declarative ones. Answer Set Programming was chosen because of its high efficiency, as witnessed by the many applications in industry. To achieve that, the FMS of BMW and Incubed IT were first analyzed, and then integrated with a new scheduler modeled in ASP. In the previous section, results and comparison between the two approaches in both companies are shown and analyzed. As we expected, there is not a clear winner between the two systems, but this comparison highlighted the pros and cons of both languages, whose performance highly depend on the kind and size of tasks to be accomplished. One main quality criterion of the FMS is the performance and the quality of the results. To evaluate the criterion, test scenarios have been set up that are based on typical use-cases of the FMS. Regarding the BMW use-case, the imperative solution is significantly faster than the declarative one, especially for the task assignment problem. However, in ASP we make use of a different optimization technique, which rewards with better solutions . This different strategy led to a trade-off between solving time and solution quality: if the imperative method is faster, ASP finds better solutions. The Incubed IT use-case gave instead different results, making clear how a very specific scenario can benefit from a particular approach rather than a general one. However, a common behavior can be seen from both BMW and Incubed IT, which represents the main weakness of ASP and enumeration tools in general. It does not scale over the size of the problem. Yet in the Incubed IT scenario in which ASP does not experience a combinatorial blow-up, it finds the best solution in less time than Java, without com-

Test Scenario	Java Implementation			ASP within Java Implementation			Standalone ASP		
	μ [ms]	σ [ms]	# TRS	μ [ms]	σ [ms]	# TRS	μ [ms]	σ [ms]	# TRS
1	278.30	226.64	10	121.30	143.50	10	71.90	14.96	10
2	491.30	223.23	10	495.00	339.16	10	278.67	236.67	10
3	2,572.80	4,894.22	10	/	/	0	/	/	0

Table 3: Runtime and solved test runs (TRS) for the different Incubed IT assignment implementations

Use-case	Imperative	ASP after 1 sec	ASP after 5 sec	ASP after 60 sec
BMW task assign.	7242	3834	3834	3834
BMW park & charge assign.	3217	6979	6530	4669
Incubed IT assign.	186	293	302	257

Table 4: Traveling costs for all the use-cases [m]

promising over the quality.

To conclude, this work has shown that declarative programming can perform well on real-world logistics scenario, especially when we are interested in the quality of the optimization. Another important advantage of this approach is the separation between the description and the solving of the problem. In fact, performance can be an issue with ASP, especially in a dynamic planning scenario, but fortunately state-of-the-art solvers like Clingo or DLV come with many meta-heuristics and optimizations to play with. Once the proper settings for the specific scenario are found, solving time can improve considerably, without having to modify the code at all. In all the cases in which a greedy algorithm is proven to perform well, in terms of both quality and solving time, imperative programming still remains the best choice.

References

[1] *Reasoning Web. Semantic Technologies for Intelligent Data Access - 9th International Summer School 2013, Mannheim, Germany, July 30 - August 2, 2013. Proceedings*, volume 8067 of *Lecture Notes in Computer Science*. Springer, 2013.

[2] M. Abseher, M. Gebser, N. Musliu, T. Schaub, and S. Woltran. Shift design with answer set programming. *Fundam. Inform.*, 147(1):1–25, 2016.

[3] C. Dodaro and M. Maratea. Nurse scheduling via answer set programming. In M. Balduccini and T. Janhunen, editors, *Logic Programming and Nonmonotonic Reasoning*, pages 301–307. Springer International Publishing, 2017.

[4] E. Erdem, E. Aker, and V. Patoglu. Answer set programming for collaborative housekeeping robotics: representation, reasoning, and execution. 5(4):275–291, 2012.

[5] W. Faber, N. Leone, and S. Perri. The intelligent grounder of DLV. In E. Erdem, J. Lee, Y. Lierler, and D. Pearce, editors, *Correct Reasoning: Essays on Logic-Based AI in Honour of Vladimir Lifschitz*, pages 247–264, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[6] F. Fabricius. ASP-based Task Scheduling for Industrial Transport Robots. Master's thesis, Graz University of Technology, 2019.

[7] M. Gebser, R. Kaminski, B. Kaufmann, and T. Schaub. Answer set solving in practice. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 6(3):1–238, 2012.

[8] M. Gebser, R. Kaminski, A. König, and T. Schaub. Advances in gringo series 3. In J. P. Delgrande and W. Faber, editors, *Logic Programming and Nonmonotonic Reasoning*, pages 345–351, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[9] M. Gebser, R. Kaminski, and T. Schaub. Grounding recursive aggregates: Preliminary report. 2016. Workshop proceeding.

[10] M. Gebser, P. Obermeier, T. Schaub, M. Ratsch-Heitmann, and M. Runge. Routing driverless transport vehicles in car assembly with answer set programming. 18(3-4):520–534, 2018.

[11] M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. In R. Kowalski, Bowen, and Kenneth, editors, *Proceedings of International Logic Programming Conference and Symposium*, pages 1070–1080. MIT Press, 1988.

[12] N. Leone and F. Ricca. Answer set programming: A tour from the basics to advanced development tools and industrial applications. In *Reasoning Web. Web Logic Rules: 11th International Summer School 2015*, pages 308–326, 07 2015.

[13] V. Nguyen, P. Obermeier, T. C. Son, T. Schaub, and W. Yeoh. Generalized target assignment and path

finding using answer set programming. *IJCAI International Joint Conference on Artificial Intelligence*, pages 1216–1223, 2017.

[14] F. Ricca, G. Grasso, V. Lio, and S. Iiritano. Team-building with answer set programming in the gioia-tauro seaport. *Theory and Practice of Logic Programming*, 12(03):361–381, 2012.

[15] Z. G. Saribatur, E. Erdem, and V. Patoglu. Cognitive factories with multiple teams of heterogeneous robots: Hybrid reasoning for optimal feasible global plans. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2923–2930, 2014.

[16] S. Schieweck, G. Kern-Isberner, and M. ten Hompel. Using answer set programming in an order-picking system with cellular transport vehicles. *IEEE International Conference on Industrial Engineering and Engineering Management*, pages 1600–1604, 2016.

[17] M. Selmair, S. Hauers, and L. Gustafsson-Ende. Scheduling charging operations of autonomous agvs in automotive in-house logistics. *ASIM*, 2019.

Learning Manipulation Tasks from Vision-based Teleoperation

Matthias Hirschmanner, Bernhard Neuberger, Timothy Patten, Markus Vincze
Automation and Control Institute, TU Wien, Vienna, Austria
{hirschmanner,neuberger,patten,vincze}@acin.tuwien.ac.at

Ali Jamadi
Ferdowsi University of Mashhad, Mashhad, Iran
a.jamadi@mail.um.ac.ir

Abstract. *Learning from demonstration is an approach to directly teach robots new tasks without explicit programming. Prior methods typically collect demonstration data through kinesthetic teaching or teleoperation. This is challenging because the human must physically interact with the robot or use specialized hardware. This paper presents a teleoperation system based on tracking the human hand to alleviate the requirement of specific tools for robot control. The data recorded during the demonstration is used to train a deep imitation learning model that enables the robot to imitate the task. We conduct experiments with a KUKA LWR IV+ robotic arm for the task of pushing an object from a random start location to a goal location. Results show the successful completion of the task by the robot after only 100 collected demonstrations. In comparison to the baseline model, the introduction of regularization and data augmentation leads to a higher success rate.*

1. Introduction

Robot manipulation tasks in domestic services and industry are highly complex due to the various system components that are necessary to achieve the goal. As a result, it is difficult to directly program robust robot manipulation strategies. Reinforcement learning is an alternative approach that alleviates the requirement for human programming and instead enables a robot platform to learn from its own experience [4, 11, 15]. However, this approach suffers from substantial training time, with some work reporting training times in the order of months [15]. Learning from demonstration (LfD) is an attractive solution in which a human illustrates how to perform a task and the robot attempts to imitate [21, 2]. This requires no human programming and far fewer training examples compared to reinforcement learning methods.

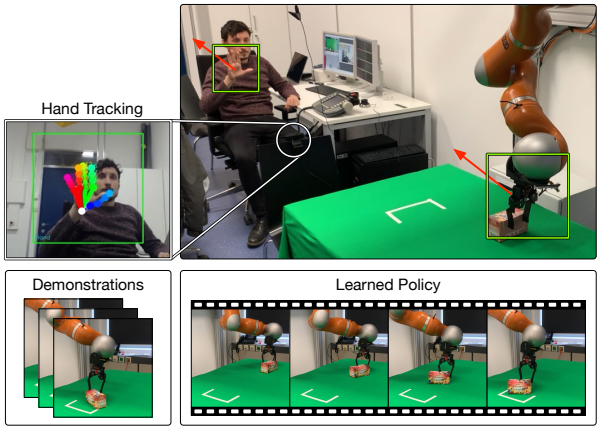


Figure 1. Teleoperating the robot arm using hand tracking from RGB images. The demonstrations are used to teach a policy to perform a task (e.g. push the box to the goal).

Demonstrations for learning are often collected through kinesthetic teaching [1] or teleoperation [27]. However, these methods are cumbersome because the human must either physically interact with the robot to generate example motions or control the robot system with specialized hardware that the operator may not have experience with. Despite the advances of teleoperation systems that enable novices to improve task performance after only a small number of attempts [8], the hardware is not always readily available. LfD can also leverage simulation [18] or by directly observing human activity [9, 13, 16, 24]. But these approaches demand additional solutions to transfer across domains.

To that end, we present an end-to-end system for LfD through vision-based teleoperation, which alleviates the necessity for virtual reality and teleop-

eration hardware while still directly controlling the robot platform to avoid the domain shift. We directly track the human hand using a webcam and use the estimated hand pose to control the end-effector of the robot. The demonstration data are used to train a neural network, based on the architecture of [27], to enable imitation by the robot system. We extend this work to include different regularization techniques during training and data augmentation to manage changes in brightness and imperfect demonstrations.

Our method is implemented for the KUKA LWR IV+ [3] robotic arm for the task of pushing objects. Experiments show that the robot is able to replicate the demonstrated task with as few as 100 recorded examples. In comparison to the baseline [27], our inclusion of regularization and data augmentation achieves a higher success rate.

In summary, we make the following contributions:

- A vision-based hand tracking system to teleoperate a robot arm to perform manipulation tasks.
- Training of a neural network with our generated teleoperated data that enables task imitation.
- Evaluation of the generalization of the imitation learning to unseen configurations.
- Improvements over the baseline by including regularization methods during the training.

The remainder of this paper is as follows. Section 2 reviews related work and Section 3 presents our approach. In Section 4 we present our experiments and results. Section 5 concludes the paper.

2. Related Work

A popular approach to program a robot to perform manipulation tasks is learning from demonstration [21, 2]. This involves recording example manipulation sequences and then to transfer the trajectories to the robot platform to perform the task itself. Trajectories are typically recorded using kinesthetic teaching [22, 19, 1], teleoperation [27, 10, 20] or generated in simulation [6, 18]. Given a set of demonstrations, these methods find an appropriate mapping in order to replicate the closest matching trajectory, often making adaptations due to the variation between the current and demonstrated scenarios. Some approaches represent the demonstrations as a set of primitives by encoding the trajectories and then generating robot motions through probabilistic methods, e.g., Gaussian mixtures [5], Gaussian processes [22] or dynamic movement primitives [19, 12]. This allows for a more efficient search for the most appropriate trajectory to replicate.

More recent works apply deep neural networks to learn visuomotor policies that map input images to robot trajectories through behavioral cloning [27, 20]. A network is trained on demonstrations to learn the image-to-action mapping such that a closed-loop controller commands the manipulator through sequences of states to complete the task. In this line of work, teleoperation is the preferred method to kinesthetic teaching because the human does not contaminate the training images.

Extensions have been made that generalize the models to multiple tasks, which allows few- or even one-shot learning of new tasks [6, 26]. These methods apply meta-learning to efficiently adapt a learned model, trained on many prior tasks, to a new task that is to be imitated. James *et al*. [10] take a different approach and use metric learning to create a task embedding. Imitating a new demonstration is achieved by training a control network to translate learned task embeddings into desired actions. Huang *et al*. [9] propose neural task graphs to learn the common structure of tasks and the conjugate relationship between observed states and actions.

Another direction of work is to learn by using only videos of humans performing tasks, e.g., [13, 16, 24]. However, human demonstrations do not provide sufficient supervision for learning. Therefore, other approaches explicitly learn the relationship between human and robot demonstrations in order to directly imitate human tasks in the online setting [26].

In this work, we build on the approaches for learning visuomotor policies through behavioral cloning. In particular, we adapt the methodology presented by Zhang *et al*. [27] by replacing the teleoperation hardware with a vision-based system. Our work is complementary to it as well as to the extension that incorporates human demonstrations [26] by using our teleoperation system as an alternative.

3. Approach

This section describes our approach for learning from demonstration, an overview is given in Figure 2. For teleoperation, a webcam is used to track the hand (Section 3.1) to generate positions that control the robot's end-effector (Section 3.2). During the trajectory, the RGB-D images from a ceiling

mounted ASUS Xtion camera are recorded with the end-effector pose of the robot. Many demonstrations are shown to create a dataset that is used to train a deep imitation learning model (Section 3.3). The learned policy is then executed by the robot using only the live RGB-D images and end-effector poses.

3.1. Hand Tracking

The hand tracking method developed by Panteleris *et al*. [17] is used to estimate the 3D pose of the human demonstrator from RGB images in real-time. This approach consists of three steps: (1) Cropping the user hand in the image, (2) passing the cropped image to a 2D joint position estimator and (3) mapping the 2D joints on a 3D hand model to recover 3D positions of the joints.

For finding an initial bounding box of the hand, a deep neural network model [25] to detect hands in real-time is applied. Afterwards, the cropped image of the hand is passed through the hand key-point localization model of [7] to estimate the 2D location of the hand joints. It localizes the 21 key-points for the wrist, 5 fingertips and $5 \times 3 = 15$ finger joints. This specific model was selected because it matches or outperforms other state-of-the-art methods but with much lower computational requirements. In the end, the 2D locations of the joints are mapped to the 3D hand model via non-linear least-squares optimization. The 3D positions are then used as the initial step for the optimization of the next frame.

The 3D positions of the joints are also used to update the bounding box of the hand, which eliminates the need to use the hand detector model for each frame. However, failure of the hand tracking module based on the hand position and movement in the previous frames (e.g. due to sudden movements, occlusion, or failure in 2D localization), results in poor optimization. Therefore, to make the tracker more robust, the optimization score is checked to reset the optimizer's initial state and to use the hand detector to find a new bounding box for the hand if necessary.

3.2. Robot Control

For the teleoperation of the robot end-effector, the 3D hand position from the hand tracking system is compared with an initial hand position. If the difference between the current and the initial position for any Cartesian coordinate is above a certain threshold κ, a new end-effector position is calculated and commanded to the robot. This difference is then transformed from the camera frame to the robot base frame and denoted as $\mathbf{h}$. The transformation aligns the directions of the hand and end-effector movement to allow intuitive teleoperation.

The desired end-effector position $\mathbf{p}^*$ is calculated by adding the current end-effector position $\mathbf{p}$ and the value $\Delta\mathbf{p}$. This is calculated for each Cartesian coordinate with $p_i, h_i \in \mathbf{p}, \mathbf{h}$ according to:

$$\Delta p_i = \begin{cases} \alpha(\min\{h_{max}, h_i - \kappa\}) & \forall\ h_i > \kappa, \\ \alpha(\max\{h_{min}, -h_i - \kappa\}) & \forall\ h_i < -\kappa, \\ 0 & \text{otherwise}, \end{cases} \tag{1}$$

where h_{max} expressing the upper, h_{min} the lower limit and α as a parameter that indirectly allows the sensitivity to speed to be tuned.

As described in Section 3.3, $\Delta\mathbf{p}$ is directly learned. When executing the learned policy, $\Delta\mathbf{p}$ is used to calculate the desired end-effector position. For either the teleoperation or the task execution by the learned policy, the desired end-effector position is updated continuously and commanded to the robot. The orientation of the end-effector could be changed similarly but was not necessary for our specific task.

3.3. Deep Imitation Learning

We employed the algorithm presented by [27] and adapted it in several ways to work with our robotic setup involving imperfect demonstrations and changing environment conditions (e.g. brightness). The adapted network can be seen in Figure 2. The input $\mathbf{o}_t$ at each timestep t consists of the cropped and scaled color image $\mathbf{I}_t \in \mathbb{R}^{120\times160\times3}$, depth image $\mathbf{D}_t \in \mathbb{R}^{120\times160\times1}$, and the 5 most recent end-effector positions $\mathbf{p}_{t-4:t} \in \mathbb{R}^{15}$. After 3 convolutional layers, the data is passed through a spatial softmax layer introduced in [14]. During training, the output of this layer is used for auxiliary predictions of the current end-effector position and the end-effector position at the end of each demonstration with two fully connected layers per auxiliary prediction. The output of the network is the change of the end-effector position of the robot $\Delta\mathbf{p}$ in millimetres. Compared to [27], we omit one convolutional layer but use more units in our dense layers, which slightly reduces training time without deteriorating performance. Since we do not change the orientation of the end-effector for this simple task, we can simplify the output of the network at time t to be $\Delta\mathbf{p}_t = \pi_\theta(\mathbf{o}_t) \in \mathbb{R}^3$.

The input data is augmented by randomly changing the brightness during training and batch normalization is added after each layer to better cope with

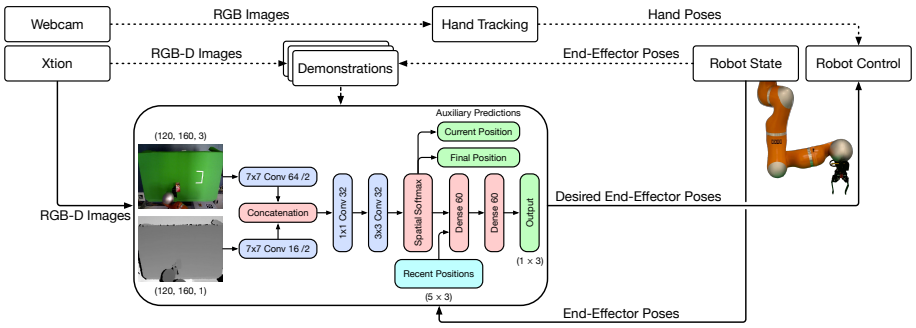


Figure 2. Overview of the system. The dashed lines show the procedure to collect demonstrations for training. The continuous lines show the information flow during policy execution.

the changing lighting conditions in the test environment. Additionally, we added dropout of the recent end-effector positions to avoid the robot following the same trajectory during most executions and not taking the object position into account.

The overall loss is defined as

$$\mathcal{L}(\theta) = \lambda_{l1}\mathcal{L}_{l1}+\lambda_{l2}\mathcal{L}_{l2}+\lambda_c\mathcal{L}_c+\lambda_s\mathcal{L}_s+\lambda_{aux}\Sigma_a\mathcal{L}_{aux}^{(a)}. \quad (2)$$

The first two terms are the $l1$ and $l2$ losses. $\mathcal{L}_c$ is the cosine loss and $\mathcal{L}_{aux}$ are the $l2$ losses of the auxiliary predictions. Compared to [27], we added the loss

$$\mathcal{L}_s = \exp\left(-||\pi_\theta(\mathbf{o}_t)||^2\right) \quad (3)$$

that penalizes very slow speeds. The weights were chosen as $\lambda_{l1} = 1.0$, $\lambda_{l2} = 0.01$, $\lambda_c = 0.05$, $\lambda_s = 0.1$, and $\lambda_{aux} = 0.01$.

4. Experiments

This section presents the experimental results. We first describe the setup and procedure for collecting demonstration data. We analyze the performance of our method with respect to the network design.

4.1. Experimental Setup

All experiments are conducted with a KUKA LWR IV+ [3] robotic arm using the provided control unit. The arm has 7 degrees of freedom and is controlled with position commands for the joints. The arm is mounted on the ceiling with a small table standing underneath it on which the target object (box) rests. The goal region is marked with tape. An ASUS Xtion RGB-D camera is mounted to the ceiling to capture the scene from above. For hand tracking, a separate webcam is used and faces the operator.

The algorithms for the hand tracking and the task execution run on a remote PC connected to the KUKA control unit via Ethernet. The communication between the remote PC and the control unit is enabled through the kuka-lwr-ros package[1] using the fast research interface (FRI) [23].

For data collection, the teleoperator directly faces the robot and the webcam. For each demonstration, the box is positioned randomly on the table. The teleoperator moves the box to the goal position using our control scheme. We collected 98 demonstrations with an average length of 42.8 s with a rate of 10 Hz for our evaluation. That is significantly above the average demonstration time per task of [27], which is between 3.7 s and 11.6 s and necessitates our changes to the architecture to deal with these imperfect demonstrations.

4.2. Results

For the evaluation, the workspace of the robot on the table is divided into a grid of 9 different positions with 20 cm intervals. Per position, the learned policy is executed for 4 different rotations of the box $(-45°, 0°, 45°, 90°)$. We measure both if the box is pushed towards the goal (*started push*) as well as if at least part of it is pushed into the goal (*success*). If the robot starts to push the box, but loses it, we restart the policy manually and keep the box in the same position when the end-effector stops or leaves the workspace. This could be automated with a simple heuristic. If the task can be achieved in a consecutive trial, we still count it as a success.

As shown in Table 1, our learned policy started to push the box in the right direction in 86.1 % of the cases and reached the goal in 58.3 % of the overall attempts. A reason for most failure cases is the grid nature of our workspace separation, which inherently tests the robot on the edges of its workspace where it is much more difficult to perform the task.

We conducted an ablation study to evaluate our changes to the original architecture of [27]. We re-

[1] https://github.com/epfl-lasa/kuka-lwr-ros

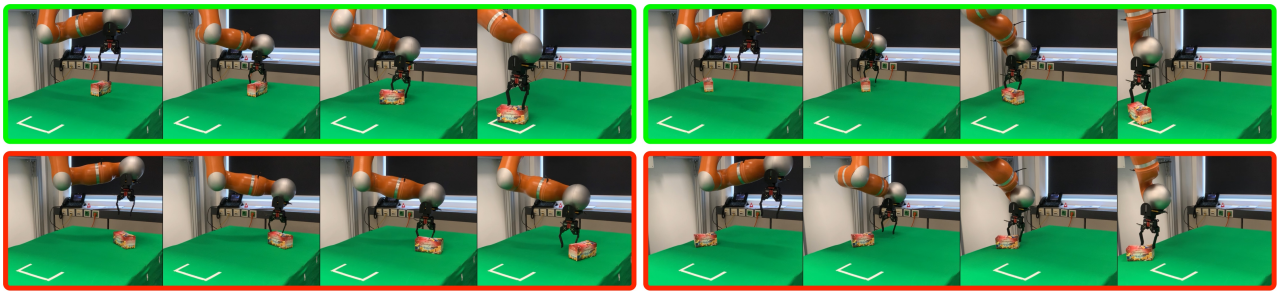


Figure 3. Examples of the learned policy. First row shows successful trials. Bottom row shows failures.

Table 1. Ablation study

	Started Push	Success
Vanilla Policy	50.0 %	27.8 %
No Dropout	66.7 %	27.8 %
No Brightness Aug.	75.0 %	41.7 %
Our Policy	86.1 %	58.3 %

implemented the original model and adapted it to our robotic platform and task. To test the effect of individual changes, we applied our policy once without dropout and once without data augmentation. The vanilla policy and our policy without dropout only achieve a success rate of 27.8 %, which were almost exclusively the trials when the box was located in a middle position and only required a straight push.

The purpose of applying dropout to the end-effector pose input of the network is to put more emphasize on the input images. With the added dropout, the success rate rises to 41.7 %. Brightness augmentation alone did not improve the overall success rate over the vanilla policy. However, the combination of dropout and brightness augmentation achieved a success rate of 58.3 %. We introduced the data augmentation due to changing lighting conditions in the test environment during the demonstrations. For the evaluation we kept the lighting conditions the same.

Qualitative results are presented in Figure 3. The first row shows sequences of successful trials in which the box is pushed to the goal. The second row shows examples of failures. In one case, the robot end-effector slides past the box and the policy loses the target. In the second case, the box is pushed to a location that is not the goal.

5. Conclusion

This paper presented an approach for learning from demonstration using a vision-based solution for robot teleoperation. A hand tracking method was employed to generate commands that control the robot's end-effector as the human operator completes a manipulation task. The set of demonstrations were used to train a deep imitation learning network that learns a policy, enabling the robot to imitate the task. Experiments showed that the introduction of regularization and data augmentation increased the success rate over the baseline method.

For future work, we plan to combine the LfD approach with reinforcement learning in simulation. By starting from the learned policy in simulation, the training time of reinforcement learning approaches can be greatly reduced. Additionally, combining real data with synthetic data collected in simulation mitigates the problem of domain adaptation of pure reinforcement learning methods. Another avenue is to use more high-level knowledge of the scene (e.g. object pose) to make the approach less susceptible to environment changes.

Acknowledgments

This research is partially supported by the Vienna Science and Technology Fund (WWTF), project RALLI (ICT15-045), the Austrian Science Foundation (FWF), project InDex (I3969-N30), and Festo AG & Co. KG.

References

[1] B. Akgun, M. Cakmak, K. Jiang, and A. L. Thomaz. Keyframe-based learning from demonstration. *The Int. Journal of Social Robotics*, 4(4):343–355, 2012.

[2] B. D. Argall, S. Chernova, M. Veloso, and B. Browning. A survey of robot learning from demonstration. *Robotics and Autonomous Systems*, 57(5):469–483, 2009.

[3] R. Bischoff, J. Kurth, G. Schreiber, R. Koeppe, A. Albu-Schäffer, A. Beyer, O. Eiberger, S. Haddadin, A. Stemmer, G. Grunwald, et al. The KUKA-DLR lightweight robot arm-a new reference platform for robotics research and manufacturing. In *Proc. of Int. Symposium on Robotics and German Conference on Robotics*, 2010.

[4] A. Boularias, J. A. Bagnell, and A. Stentz. Learning to manipulate unknown objects in clutter by reinforcement. In *Proc. of AAAI Conf. on Artificial Intelligence*, pages 1336–1342, 2015.

[5] S. Calinon, P. Evrard, E. Gribovskaya, A. Billard, and A. Kheddar. Learning collaborative manipulation tasks by demonstration using a haptic interface. In *Proc. of Int. Conf. on Advanced Robotics*, pages 1–6, 2009.

[6] Y. Duan, M. Andrychowicz, B. Stadie, O. Jonathan Ho, J. Schneider, I. Sutskever, P. Abbeel, and W. Zaremba. One-shot imitation learning. In *Advances in Neural Information Processing Systems 30*, pages 1087–1098, 2017.

[7] F. Gouidis, P. Panteleris, I. Oikonomidis, and A. Argyros. Accurate hand keypoint localization on mobile devices. In *Proc. of IEEE Int. Conf. on Machine Vision Applications*, 2019.

[8] M. Hirschmanner, C. Tsiourti, T. Patten, and M. Vincze. Virtual reality teleoperation of a humanoid robot using markerless human upper body pose imitation. In *Proc. of IEEE-RAS Int. Conf. on Humanoid Robots*, 2019.

[9] D. Huang, S. Nair, D. Xu, Y. Zhu, A. Garg, L. Fei-Fei, S. Savarese, and J. C. Niebles. Neural task graphs: Generalizing to unseen tasks from a single video demonstration. In *Proc. of IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, pages 8557–8566, 2019.

[10] S. James, M. Bloesch, and A. J. Davison. Task-embedded control networks for few-shot imitation learning. In *Proc. of Conf. on Robot Learning*, pages 783–795, 2018.

[11] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke, and S. Levine. Scalable deep reinforcement learning for vision-based robotic manipulation. In *Proc. of Conf. on Robot Learning*, pages 651–673, 2018.

[12] G. Konidaris, S. Kuindersma, R. Grupen, and A. Barto. Robot learning from demonstration by constructing skill trees. *The Int. Journal of Robotics Research*, 31(3):360–375, 2012.

[13] V. Krüger, D. L. Herzog, S. Baby, A. Ude, and D. Kragic. Learning actions from observations. *IEEE Robotics Automation Magazine*, 17(2):30–43, 2010.

[14] S. Levine, C. Finn, T. Darrell, and P. Abbeel. End-to-end training of deep visuomotor policies. *J. Mach. Learn. Res.*, 17(1):13341373, Jan. 2016.

[15] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *The Int. Journal of Robotics Research*, 47(4-5):421–436, 2018.

[16] Y. Liu, A. Gupta, P. Abbeel, and S. Levine. Imitation from observation: Learning to imitate behaviors from raw video via context translation. In *Proc. of IEEE Int. Conf. on Robotics and Automation*, pages 1118–1125, 2018.

[17] P. Panteleris, I. Oikonomidis, and A. Argyros. Using a single RGB frame for real time 3D hand pose estimation in the wild. In *Proc. of IEEE Winter Conf. on Applications of Computer Vision*, pages 436–445, 2018.

[18] A. Pashevich, R. Strudel, I. Kalevatykh, I. Laptev, and C. Schmid. Learning to augment synthetic images for Sim2Real policy transfer. In *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 2651–2657, 2019.

[19] P. Pastor, L. Righetti, M. Kalakrishnan, and S. Schaal. Online movement adaptation based on previous sensor experiences. In *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 365–371, 2011.

[20] R. Rahmatizadeh, P. Abolghasemi, L. Blni, and S. Levine. Vision-based multi-task manipulation for inexpensive robots using end-to-end learning from demonstration. In *Proc. of IEEE Int. Conf. on Robotics and Automation*, pages 3758–3765, 2018.

[21] S. Schaal. Is imitation learning the route to humanoid robots? *Trends in cognitive sciences*, 3(6):233–242, 1999.

[22] M. Schneider and W. Ertel. Robot learning by demonstration with local Gaussian process regression. In *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 255–260, 2010.

[23] G. Schreiber, A. Stemmer, and R. Bischoff. The fast research interface for the KUKA lightweight robot. In *IEEE ICRA 2010 Workshop on Innovative Robot Control Architectures for Demanding (Research) Applications – How to Modify and Enhance Commercial Controllers*, 2010.

[24] P. Sermanet, C. Lynch, Y. Chebotar, J. Hsu, E. Jang, S. Schaal, and S. Levine. Time-contrastive networks: Self-supervised learning from video. In *Proc. of IEEE Int. Conf. on Robotics and Automation*, pages 1134–1141, 2018.

[25] D. Victor. Handtrack: A library for prototyping real-time hand tracking interfaces using convolutional neural networks. *GitHub repository*, 2017.

[26] T. Yu, C. Finn, S. Dasari, A. Xie, T. Zhang, P. Abbeel, and S. Levine. One-shot imitation from observing humans via domain-adaptive meta-learning. In *Proc. of Robotics: Science and Systems*, 2018.

[27] T. Zhang, Z. McCarthy, O. Jow, D. Lee, X. Chen, K. Goldberg, and P. Abbeel. Deep imitation learning for complex manipulation tasks from virtual reality teleoperation. In *Proc. of IEEE Int. Conf. on Robotics and Automation*, pages 5628–5635, 2018.

Reactive Motion Planning Framework Inspired by Hybrid Automata

Csaba Hajdu, Áron Ballagi
Széchenyi István University
{hajdu.csaba,ballagi}@sze.hu

Abstract. *This paper presents a motion planning framework controlled by reactive events and producing feedback data suitable to be processed by various learning and verification methods (e.g. reinforcement learning, runtime monitoring). Our architecture decomposes subtasks of motion planning into separate perception and trajectory planner parts. In our architecture, we interact between these distributed parts through discrete-timed events controlled by timed state machines, besides classical continuous state flow. Our research primarily focuses on autonomous vehicle research, so this framework is supposed to satisfy the requirements of this field. The motion planner framework interfaces a widely-used robotic middleware.*

1. Introduction

Motion planning (or trajectory planning) is a mandatory task both in mobile robotics and in autonomous vehicle navigation [4]. The field has been actively researched and used, providing efficient algorithms suitable for different domains and robot setups. The role of motion planning in robotics is to create a feasible, collision-free path between the location of the agent (mobile robot or vehicle) and an arbitrarily defined goal point, based on the agent's sensory input and actuation. On the other hand, the emergence of autonomous vehicles and other special UGVs requires high-reliability, computational efficiency and optimization of velocity profile even in rough environmental conditions.

The typical problems of motion planner frameworks are their relatively hard extension and limited verification capabilities. In this paper, we propose a prototype of a motion planning architecture with the focus on providing comprehensive verification output and extension capabilities.



Figure 1. Electronically modified autonomous test vehicle

2. Motivation and related work

The development of a new motion planning framework was motivated by ongoing research at our university. We are developing an autonomous vehicle (an electronically modified Nissan Leaf equipped with numerous sensors, Figure 1) and a differential drive robot in various projects. Both rely on motion planning, thus our aim is to create a motion planner framework usable in both application - with minimal configuration effort.

Many commercially available unmanned ground vehicles (UGV) use ROS and its integrated navigation component, *move_base*. This framework is a monolithic implementation with plugin-oriented extension and occupancy grids as a basis of environment representation. Some of these issues had been addressed in *move_base_flex* [5]. In autonomous vehicle frameworks, Autoware [3] provides a loose architecture enabling the replacement of its built-in motion planning component with different solutions. In both approaches, reactive events (e.g. synchronization of all incoming topics, the transition to replanning, etc.) in both systems are relatively hard to trace and debug.

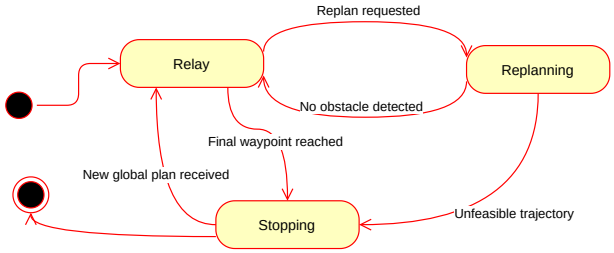


Figure 2. Overview of the local planner state machine described as a state machine

3. Architecture proposal

In this section, we propose the architecture of our motion planner framework[1]. By investigating other planner frameworks we found the following typical characteristics:

1. The underlying planner is controlled by events (e.g. transition to re-planning, recovery initiation) in a fashion of a state-machine based approach (a brief overview of the local planner state machine is shown on Figure 2).

2. Execution should not start before all input information has been received recently.

3. Transitions between states are not necessarily instantaneous, requiring smoother switching between behaviors.

The development of a (hybrid) timed state machine library was motivated by these properties, especially to efficiently resolve property 1 and 2. Hybrid automata [1] is a well-studied way to model and verify systems with both discrete and continuous timed properties and also to describe robot behaviors [2]. A behavior similar to what is presented on Figure 2 can be easily mapped to hybrid automata formalism. Transitions are either governed by discrete events and continuous activities. For example, a continuous variable in this case could be the distance to the closest obstacle detected and a typical discrete event is the request to replan a segment of the trajectory or to execute fallback scenario. We ensure, that each transitions are published to a middleware framework, enabling versatile runtime verification.

Our goal was to follow a highly-distributed architecture, where sub-tasks are decomposed from the planner component. In our approach, the perception related tasks like obstacle detection and classification are decomposed from other specific planner tasks. This enables the reuse of components and isolated verification. Perception components are interacting with planner components by inducing discrete events and modifying continuous signals. For instance, an obstacle detection component may trigger the local planner to replan by raising a discrete-timed event. After the obstacle is avoided, the planner restores the remainder of the original trajectory in relay mode.

[1] Available at https://github.com/kyberszittya/hotaru_planner.git

4. Conclusion

In conclusion, we provided an overview of a new motion planning framework under development which can be easily extended with new algorithms and tuned to specific domain requirements. A new initial motion planner framework version is created. The extension of our framework with various local planner methods is a primary focus. Global trajectory planner methods will be integrated in the future. Our automata framework and the related code-generator tool will be also enhanced.

5. Acknowledgments

This work was supported by the Hungarian Government and the European Union within the frames of the Széchenyi 2020 Programme through grant GINOP-2.3.4-15-2016-00003

References

[1] R. Alur, C. Courcoubetis, N. Halbwachs, T. A. Henzinger, P.-H. Ho, X. Nicollin, A. Olivero, J. Sifakis, and S. Yovine. The algorithmic analysis of hybrid systems. *Theoretical Computer Science*, 138(1):3 – 34, 1995.

[2] M. Egerstedt, K. Johansson, J. Lygeros, and S. Sastry. Behavior based robotics using regularized hybrid automata. In *Proceedings of the IEEE Conference on Decision and Control*, volume 4, pages 3400 – 3405 vol.4, 1999.

[3] S. Kato, S. Tokunaga, Y. Maruyama, S. Maeda, M. Hirabayashi, Y. Kitsukawa, A. Monrroy, T. Ando, Y. Fujii, and T. Azumi. Autoware on Board: Enabling Autonomous Vehicles with Embedded Systems. In *2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS)*, pages 287–296, Apr. 2018.

[4] S. M. LaValle. *Planning Algorithms*. Cambridge University Press, New York, NY, USA, 2006.

[5] S. Pütz, J. S. Simón, and J. Hertzberg. Move Base Flex: A Highly Flexible Navigation Framework for Mobile Robots. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2018.

Automated Log Ordering through Robotic Grasper

Stephan Weiss[1], Stefan Ainetter[2], Fred Arneitz[1], Dailys Arronde Perez[1], Rohit Dhakate[1], Friedrich Fraundorfer[2], Harald Gietler[1], Wolfgang Gubensäk[3], Mylena Medeiros Dos Reis Ferreira[1], Christian Stetco[1], Hubert Zangl[1]

[1)]University of Klagenfurt
{[first given name].[first surname]}@aau.at

[2)]Graz University of Technology
{stefan.ainetter, fraundorfer}@icg.tugraz.at

[3)]Springer Maschinenfabrik GmbH
Wolfgang.Gubensaek@springer.eu

Abstract. *This work focuses on retrofitting a crane model in the wood industry for automated log grasping. AI inspired vision based approaches are used to categorize and segment the logs and their geometry to subsequently define optimal grasping poses. Retrofittable sensors and robust control strategies for cost efficient upgrading of existing manually operated cranes towards autonomous systems are developed.*

1. Introduction

Classical production lines and handling processes for raw materials often have a long history and incorporate a large amount of experience based knowledge for process optimization and handling routines. Nowadays, these processes seem to be stuck in a local minima in terms of efficiency and performance due to human factors. With the available degree of automation, robustness of AI based perception and decision making, and novel sensor technology, a re-thinking of these well established processes can take place. Instead of a radical approach to replace existing infrastructure, this work leverages currently installed machines in the wood sector and enables them to work autonomously through retro-fitting of sensors, autonomy, and AI based scene understanding. The project has a strong focus on bringing advanced methods in the corresponding research fields to practice. Hence, a model log crane was built as a 1:5 scaled down copy of a real log crane (Fig. 1).

* The research leading to these results has received funding from BMVIT under grant n. 864807 (AutoLOG)

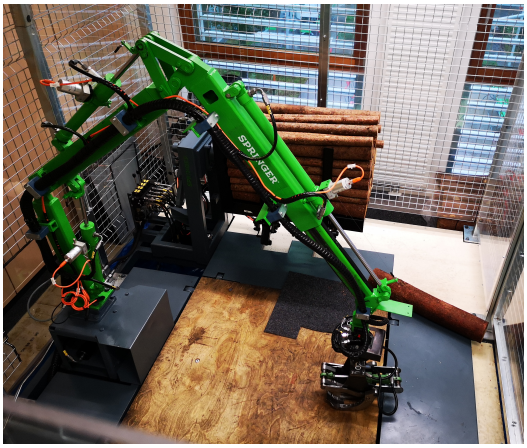


Figure 1. Crane model in 1:5 scaled version of a real crane used in the wood sector. The hydraulics are specifically designed to match this scale. Manual control is identical to the real versions.

2. Model and Retrofittable Sensor Design

The 1:5 scaled crane model has been designed and manufactured from scratch to match the properties of the real counterparts. This includes hydraulic actuators, end-effector with two free joints and an actuated revolute joint with unconstrained 360° actuation, and backlash. For tests and evaluation, we installed wire-rope sensors on the hydraulic pistons to measure their current position. Novel capacitive and inductive sensors have been designed and implemented as described in Section 2.1 to measure the current absolute angles and to provide feedback on the grasping quality. Apart of the crane itself, the overall system (Fig. 2) also contains a log storage box with automated emptying mechanism. Emptying

the box is done by asynchronously opening the box such that the model logs spread randomly on the floor. The floor area designed as log picking area can be shielded during a box emptying process to prevent the logs from spreading too wide in the area. With the project goal of the crane being able to autonomously store the logs in the box, this automated emptying process enables an endless cycle for automated training refining the AI based procedures without supervision.

The crane is controlled at a high level by an external PC which is connected via Ethernet to a HAWE-ESX control unit. The ESX controls the hydraulic pistons and sends the signals of the wire-rope and custom angular sensors via Ethernet back to the host PC. The PC also receives data from two cameras mounted on the fix and movable part of the crane as well as from five IMUs mounted on each of the crane joints. These sensors will serve for automated model creation as we assume to not have CAD drawings of every crane in a retrofitting process. The overall connectivity schematic is shown in Fig. 3.

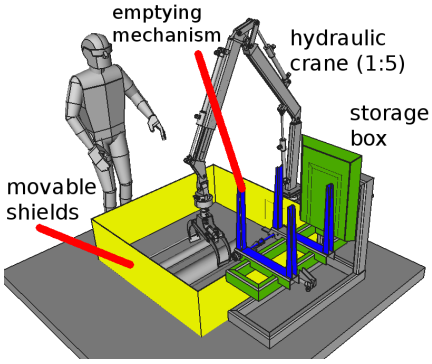


Figure 2. Crane model system with automated elements for continuous learning without human intervention.

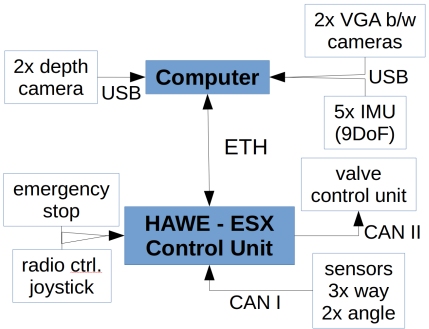


Figure 3. Overview on the connectivity of the model crane, the sensors, and the external PC.

2.1. Retrofittable Sensors

Automating machinery in the wood sector is challenging since not only the sensors that enable autonomy need to be equipped ideally without disassembling the machine, they also need to be autarkic in terms of energy, and withstand very harsh environments. Thus, robust magnetic angular position sensors following [1] suitable for retrofitting and wireless operation have been integrated on the crane model. They can easily be adapted for different joint geometries. The basic architecture is shown in Fig. 4 together with the lab setup (currently with wired CAN). In addition, capacitive sensors

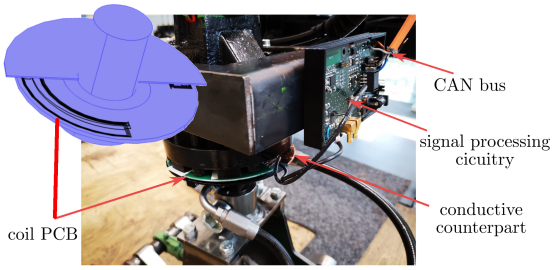


Figure 4. The experimental sensor and CAD coil geometry on the rotary joint of the end-effector: The coil PCB and signal processing circuitry is mounted to the non-rotating head whereas the conductive plate is mounted on the rotating shaft. The conductive counterpart consists of a 3D printed holder and wrapped copper foil.

following [2] are integrated in the end-effector to augment the machinery with a sense for log grasping quality (Fig. 5). The crane and sensors are simulated

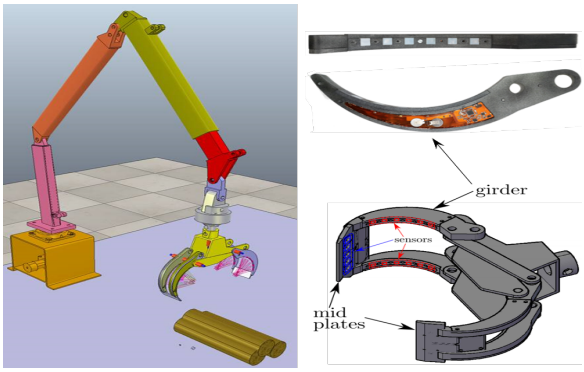


Figure 5. Left: V-REP model. Bottom right: gripper design. Top right: photograph of the gripper prototype including the sensor elements wireless electronics.

in V-REP. There, the communication and control are tested using V-REP/ROS and V-REP/Python bridge. The simulation also serves as an environment for AI training of the crane controls and for optimizations on sensor placement following [3]. A video of the simulation framework can be found in [4]

3. Control and State Estimation

The manipulator as a forest crane is vastly different compared to a standard industrial robot: the rather unconventional design requires detailed geometrical knowledge to derive the kinematic model. Also, the hydraulic driving system suffers from heavy vibrations, backlashes and jerks which require detailed dynamic parameters for proper modelling. To capture the complex relationships on existing machines where CAD and dynamic models are rarely available, we use machine learning techniques for the prediction of kinematics and dynamics parameters. Additional inertial and visual sensors further help to re-fine the overall state estimation including the adaptive estimation of the dynamic parameters defining the sway-motion of the two free joints on the end-effector. This adaptive estimation of the kinematic and dynamic parameters allows a simplified manipulator model for adaptive control schemes when picking and placing logs with sway motion.

3.1. Automated Grasping Point Prediction

To find the optimal points for the gripper to grasp a log (or logs), it is necessary to recognize graspable objects in the surrounding area of the robotic manipulator and calculate possible candidates. A candidate is defined as a point/area of the log which can successfully be grasped by the gripper. A ZED camera is used for image acquisition and consists of a stereo camera system capturing high resolution RGB-D images from the scene. Core component of the prediction method is a deep learning approach using a Convolutional Neural Network to predict grasping candidates in 2D image space, similar to [5]. The depth information is used for: 1) Automatic annotation of training data for a deep neural network by leveraging sequential depth data. This method is a step towards continuous learning making it easily possible to generate new ground truth training data during real time system application. 2) Calculation of the final 3D position of the grasping point from the previously predicted 2D grasping candidate. Fig. 6 shows a sample scenario with some logs remaining in the picking area and marked grasping locations by the AI method.

3.2. Conclusion and Next Steps

We proposed a mechanical setup for training a crane model of the wood industry for automated log grasping. The setup allows automated operation such that continuous learning without human intervention can be possible. Retrofittable sensors allow additional sensing capability in order to autonomously control the grasping procedure and to verify correct picking of the desired logs. The current results show that while the alignment of the desired gripper positions to grasp a log is correctly predicted by the AI, not all suggested locations are ideal in view of the center of gravity. Next steps will include the feedback of the capacitive sensors to correct the AI decision in an automated learning procedure.



Figure 6. Model logs in the picking area of the 1:5 scaled model crane with marked grasping positions by our AI based method. Red marks the desired locations of the two grippers in the end-effector of the crane.

References

[1] H. Gietler, C. Stetco, and H. Zangl, "Scalable retrofit angular position sensor system," in *IEEE International Conference on Instrumentation and Measurement*, Dubrovnik, May 2020.

[2] L. Faller, C. Stetco, and H. Zangl, "Design of a novel gripper system with 3d- and inkjet-printed multimodal sensors for automated grasping of a forestry robot," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Nov 2019, pp. 5620–5627.

[3] T. Mitterer and H. Zangl, "Beyond pure sensing: Ieee 21450 in digitalization of the development cycle of smart transducers," *IEEE Instrumentation & Measurement Magazine*, April 2020.

[4] Sensors for automated grasping of forestry robots. www.youtube.com/watch?v=B1S46LqfG48.

[5] F.-J. Chu, R. Xu, and P. A. Vela, "Real-world multiobject, multigrasp detection," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3355–3362, 2018.

Introducing a Morphological Box for an Extended Risk Assessment of Human-Robot Work Systems Considering Prospective System Modifications

Titanilla Komenda
Fraunhofer Austria Research GmbH
titanilla.komenda@fraunhofer.at

Martin Steiner
TÜV AUSTRIA SERVICES GMBH
martin.steiner2@tuv.at

Michael Rathmair, Mathias Brandstötter
Joanneum Research, Institute for Robotics and Mechatronics
{michael.rathmair,mathias.brandstoetter}@joanneum.at

Abstract. *The concept of human-machine collaboration is regarded as key enabler for agile production systems as collaborative robots offer new forms of flexibility. Due to inherent safety functionalities, these robots can operate without physically separating safety devices and thus provide flexibility in task allocation and execution. However, changes on the work system require a new risk assessment due to the present normative regulations, which is a tedious task as feasible changes are usually not considered in the implementation phase. This paper presents the impact of modifications on collaborative robotic cells and how they influence the risk assessment. Furthermore, a method of considering work system variants based on desired future modifications is presented so that implications can be already identified in an early design phase of the system.*

1. Introduction

Robot safety constitutes a key factor in human-robot working systems [1]. Currently, every manufacturer or integrator of a collaborative robotic application must place its application on the market in accordance with Directive 2006/42/EC (Machinery Directive) of the European Parliament and of the Council Among other things, this stipulates that the basic safety and health protection requirements listed in Annex 1 of the Machinery Directive must be met. Annex 1 of the Machinery Directive, under *General principles*, and also the ISO 10218:2012 standard requires that the manufacturer of a machine or his authorised representative must ensure that a risk assessment is carried out. This ensures that the safety and health protection requirements applicable to the machine are determined and that the machine is designed and built taking into account the results of the risk assessment. In this process of risk estimation and risk reduction, the limits of the machine, the intended use and the reasonably foreseeable misuse are determined.

In practice, EN ISO 12100:2010 (Safety of machinery - general principles for design - risk assessment and risk reduction) is often used as a method of carrying out a risk assessment. Using this methodology, the hazards that can arise from the machine are identified. The associated hazardous situations and the risks are estimated by also considering their severity of possible injuries or damages to health and the probability of their occurrence. The risks are then assessed to determine whether a risk reduction measure in accordance with the objectives of the Machinery Directive is necessary. If so, the hazards are eliminated or reduced by applying protective measures while in some circumstances organizational measures might be necessary. However, ISO 12100 is a type A standard meaning that methodologies described in its content are applicable for a very wide range of machinery and not necessarily specific to the application of robotic applications. Thus, EN ISO 10218-2, a type C standard, is in place. Section 4 of this document gives a risk assessment scheme that is specifically refined for robotic applications (under consideration of ISO 12100 and other related standards). Topics such as the design, manufacture, installation, operation, maintenance and decommissioning of the industrial robot system or cell are addressed. The basic hazards and hazardous situations

for these systems are identified and requirements are defined to eliminate or sufficiently reduce the risks associated with these hazards.

The structured process from the Machinery Directive down to the EN ISO 10218-2 shows that every safety-relevant change of the application requires a renewed risk assessment, unless this has already been considered in the original assessment. New risk assessments on an already existing work system might make required modifications impossible due to limited flexibility in the design or re-design. However, in order to consider safety-related changes in the original assessment, prospective modifications and thus system variants have to be considered in an early design phase. For this approach, however, the link between modifications and safety-related aspects is not yet clearly explored.

Even though, the Technical Specification for collaborative robotic applications, ISO/TS 15066:2016, presents a correlation of the applied robot's safety mode and the system's respective safety-related changes, the safety mode is only one of many safety-relevant modification dimensions within human-robot work systems [2]. Further, it shows drawbacks in applying the proposed safety measures, especially when integrating heavy industrial robots or sharp objects or estimating the human approach velocity [3]. Additionally, there is no advice considering the robot's movement predictability due to collision avoiding path planning or varying task allocation patterns.

As safety modes might not be an appropriate classification scheme for the identification of safety-relevant changes, new classifications schemes have been introduced, such as in [4, 5]. However, after an extensive literature review, [6] came to the conclusion that classification schemes for collaborative human-robot work systems are not applied consistently, which may lead to an incorrect identification of safety-relevant changes. To counteract this, model-based approaches have been developed either based on formal mathematical models, such as in [7], or based on simulation models, such as in [8]. A risk management simulator was for example introduced in [9], whereas [10] introduced a task-based characterization of human-cobot safety. Further, a metric depending on the distance between robot and human as well as the robot's structure was introduced in [11].

However, none of the proposed approaches considers a mutual influence of modifications on safety. In this sense, a structured procedure with the aid of an Morphological Box (MB) was developed and is described in detail in the following paper. The proposed approach should support manufacturers and system integrators in the consideration of safety-relevant changes in an early design phase of the planned collaborative human-robot work system including its prospective modifications.

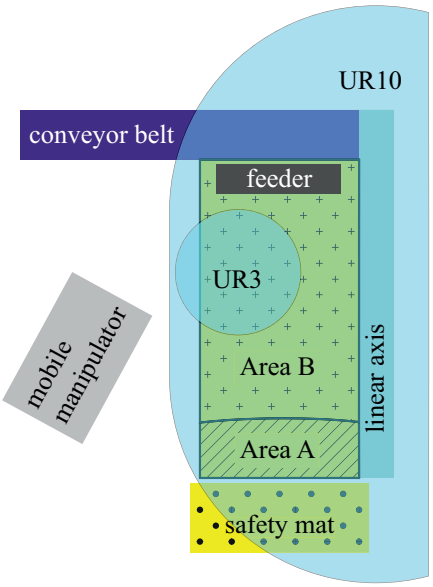


Figure 1. Structure of the workplace.

2. Impact of Modifications

Within the DR.KORS project on dynamic reconfigurability of collaborative robotic systems, 50 dimensions of modifications were identified directly influencing the safety of a human-robot system. The modification dimensions can be classified in workpiece, end effector, contact points (between human and robot), speed / acceleration, task / workflow and operating conditions / change of place. The impact of modification dimensions will be presented on a laboratory use case example for assembling rocker levers.

2.1. Use Case Description

In the laboratory use case rocker levers consisting of three separate components are assembled with a collaborative human-robot work system. Adjusting bolts are mounted on two separate rocker levers which are then assembled on a trestle. At this point, the positioning of the rocker levers on the trestle needs manual dexterity as the components tilt easily. Rocker levers and trestles are provided either by feeders or on a conveyor belt. The manipulation,

screwing and storing tasks are allocated between four resources, i.e. human, two robots and a mobile manipulator. The work system is designed in a way, that the position of peripheral appliances is variable and safety devices can be changed. Hence, the collaborative work system consists of (a) a Universal Robot UR10 on a linear axis for workpiece handling, (b) a feeder and a conveyor belt for workpiece supply, (c) a Universal Robot UR3 for workpiece assembly, (d) a human operator for workpiece assembly and workpiece removal, (e) a mobile manipulator for workpiece manipulation and (f) external safety devices, such as light curtains and laser scanners, for person safeguarding. See Figure 1 for the layout sketch of the workplace.

2.2. Modification Dimensions in the Use Case

The laboratory use case offers the possibility of 13 modification dimensions, which either influence the layout, the task allocation or the motion parameters of the involved resources.

- **Product:** Two different products can be assembled on the work system - either in mixed or unmixed production. The change of the assembled product influences the workpiece supply, the task allocation as well as the motion paths of the robots.
- **Position during collaboration (end effector height):** The end effector height indicates the position where human and robot assemble the product at the same time. It can be changed according to the ergonomic height of the operator.
- **Position during collaboration (robot base position):** The Universal Robot UR10 is mounted on a linear axis so to easily change its base position. This might be necessary due to reachability reasons when the layout of the work system changes or when new collision points arise on the anticipated robot paths due to changes in the task allocation. A choice of the robot's base position during the collaboration is possible and influences the sensitivity and stiffness of the arm due to the according robot posture.
- **Resource allocation for trestle feed:** The supply of workpieces can either be implemented in terms of a feeder, directly coming from the previous manufacturing machine on a conveyor belt or by a human operator. The change of the supply unit influences not only the layout of the work system but also the robot paths and optionally the resource allocation (depending on the picking requirements).
- **Resource allocation for screwing:** The screwing process can be either done by the Universal Robot UR3, by the human or by the mobile manipulator. A change in resource allocation for a specific task influences the temporal and spatial proximity of humans and robots and thus may influence the safety concept.
- **Safety function:** The safety function can either be implemented as force limitation or as distance monitoring. Based on the safety function, the safety devices are defined as well as the layout of the work system in terms of space requirements and the motion behaviour of the resources.
- **Type of safety device for distance monitoring:** The distance monitoring can either be implemented by a separating safety fence or by non-separating safety devices such as light curtains, laser scanners, safety mats, software-based workspace limitations or a combination of those.
- **Position of safety device for distance monitoring:** Depending on the type of safety device and the space requirements of the work system, the mounting distance of the safety devices is defined and thus the velocity of the robots. In general these safeguarding distances are regulated by the standard ISO 13855:2010 - Positioning of safeguards with respect to the approach speeds of parts of the human body.

Modification dimensions lead to system variants of a use case which are necessary for the flexibility of a collaborative work system. In order to already consider desired variants of a human-robot work system in the planning and design phase, a Morphological Box is introduced.

3. Morphological Box

A morphological analysis is a creative heuristic method introduced by the Swiss astrophysicist Fritz Zwicky which is mostly applied for fully understanding complex problem areas and considering all possible solutions without prejudice [12]. The resulting

Modification dimension		Parameter value				
Product		A	B	A&B		
Position during collaboration - end effector height		$h_{min} \leq h_{col} \leq h_{max}$				
Position during collaboration - robot base		$p_{min} \leq p_{base} \leq p_{max}$				
Robot velocity		$v_{min} \leq v_{rob} \leq v_{max}$				
Resource allocation trestle feed		human	by UR10 from feeder	by UR10 from conveyor belt		
Resource allocation insert screw		human	screwdriver	UR10	CHIMERA	
Resource allocation tighten screw		human	screwdriver	UR3		
Area A	Safety function	force limitation	distance monitoring			
	Type of safety device for distance monitoring	safety mat	light curtain	laser scanner	software	CHIMERA laser
	Position of safety device for distance monitoring	$< d_{safe}$	$\geq d_{safe}$			
Area B	Safety function	force limitation	distance monitoring			
	Type of safety device for distance monitoring	safety fence	light curtain	laser scanner	software	CHIMERA laser
	Position of safety device for distance monitoring	$< d_{safe}$	$\geq d_{safe}$			

Table 1. Morphological box indicating modification dimensions in the lab use case.

solutions are aggregated in a so called Morphological Box (MB) representing specific attributes and their individual characteristics. This multi-dimensional matrix maps all possible solutions by combining one characteristic for each attribute.

With the assistance of a MB, a far-reaching risk assessment can be carried out to clarify whether a new risk assessment (or even a new risk estimation) must be carried out when modifying the robot system. To be able to make this decision, a distinction must be made between changes that have been considered in advance and changes that have not yet been assessed.

One possibility for a considered change can be, for example, the storage area, which was defined in advance as an area and does not focus on the required storage point as is traditionally the case. This enables the MB to check whether the changed placement point is within these defined limits by comparing coordinates and to provide the operator with clear information as to whether a new risk assessment is necessary. The maximum safe speed can be used as a further example. During the application definition, the considered speed is not the one required for the process, but the maximum safe speed. This has the advantage that a change can be evaluated using the MB with the additional parameters that are now available. These two examples show, that already during planning and integration the safety assessment must be implemented in the process via the MB to be able to make practical comparisons in everyday life. Furthermore, it becomes apparent that simple changes can be clarified clearly and efficiently, whereas complex changes require a thorough examination using mathematical models that support the MB. An example with a much higher degree of complexity is the change of a possible contact point between humans and robots. These must be verified and validated in accordance with ISO/TS 15066:2016 point 6 or tested and measured in accordance with EN ISO 10218-2:2012 Annex G.

Due to the complexity of such changes, the MB can be used to conclude that a new risk assessment is required. During such a reassessment, the MB can provide support by showing specific dependencies that need to be considered for the reassessment in the risk assessment, e.g. the system limits of the gripping technology, the change in the permissible force/pressure values due to the shifting of the contact between human and machine. When using the MB, such restrictions can only be prevented by determining all relevant modifications in the planning phase to cover the broadest possible range.

In this sense, the MB presented in this paper indicates desired and possible modification dimensions for a specific use case. Here, the modification dimensions represent the attributes while the parameter values represent the characteristics. By combining specific parameter values for each modification dimension, different system variants of the use case can be defined. In contrast to conventional morphological

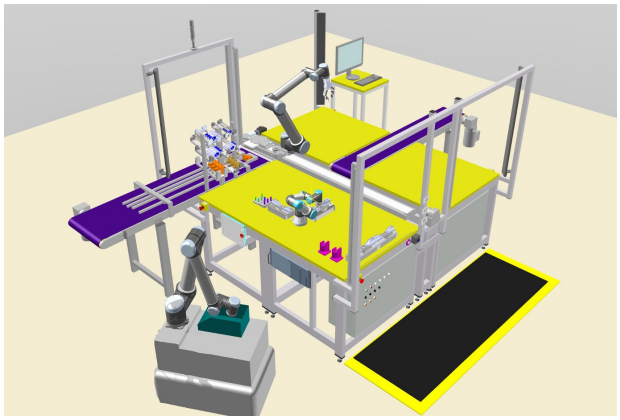


Figure 2. Simulation of the lab use case in ema Work Designer.

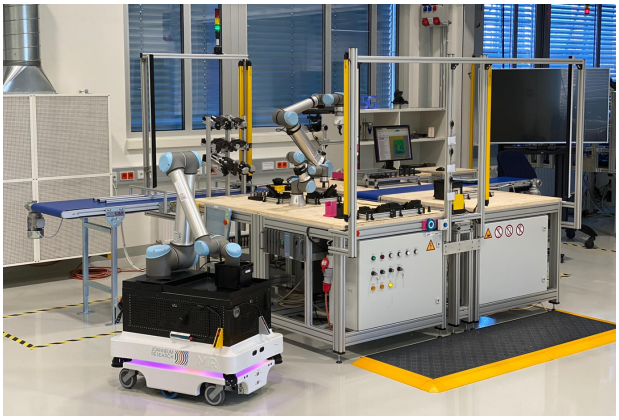


Figure 3. Setup of the use case in lab environment.

boxes, the MB in Table 1 allows for multiple selections within specific modification dimensions. In this example, the type of safety device for distance monitoring allows for combining different devices for one system variant.

For example, one use case variant could be defined as follows: The product type A of the rocker lever is assembled in the work system. The collaborative task of positioning the levers on the trestle is done in an ergonomic height for an operator. Trestles are supplied by a feeder and manipulated by a UR10. The insertion task of the adjusting bolts is carried out by the human while the tightening task is done by the UR3. The velocities of both robots is set to 500 mm/s. The safety function in both areas A and B is based on force limitation and distance monitoring by software-based workspace limitations. The distance between robot base and operator should be as large as possible during the collaboration.

3.1. Represented System Variants

The main effects on personal safety resulting from the selection of system parameters via MB are described in the following.

Impact of Resource Allocation The modification space related to the given resources spans all possibilities between manual processing to an almost fully automated scenario. Special safety considerations are relevant for those cases where a robot is allocated to a task. For this purpose, all boundary cases must be evaluated separately for e.g. critical contact situations, safety distances as well as force and pressure impacts on the involved body regions of the human. This can lead to restrictions which are stored as a set of rules for a partially automated assessment.

Impact of Safety Device Several extrinsic safety devices listed in Table 1 are exchangeable, e.g. whether a light curtain or a laser scanner is used for distance monitoring is usually irrelevant. However, the plane used to determine the safety distance has a significant influence on permissible distances and the velocity to the moving robot. Horizontal measuring safety devices, such as a safety mat or a laser scanner on a mobile manipulator, have a substantially different information content than vertical devices such as a light curtain. In contrast to horizontal safety devices, vertical safety devices have a higher uncertainty in determining the location of humans. However, a safety mat can be partially skipped by a human, whereas a laser scanner mounted on a mobile robot system, can be used in variable locations.

Impact of Workpiece Supply The feeding of the workpieces mainly influences mechanical safety characteristics, which can be determined by means of a risk assessment. Therefore, the type of the feeding system has no significant effect on the safety related system variant.

3.2. Simulated and Experimental Setup

The virtually designed and simulated laboratory use case is shown in Figure 2, whereas the physical setup of the use case is shown in Figure 3. All required modification dimensions were taken into account, which gives the impression that unnecessary redundancies exist especially for the listed external safety devices. However, these allow us to perform specific studies on meaningful combinations of safety devices and a direct and detailed comparison between them.

In order to assess the effects of modifications on

the system, different configurations of the setup are analyzed. Quantitative differences, such as cycle and operating times of the resources are obtained by simulations in ema Work Designer. In order to validate the presented method for safe system modification, four different variants were implemented on the real plant to cover a wide range of variation possibilities. The parameters of six modification dimensions were varied, specifically the product, the resource allocation (trestle feed, insert screw, tighten screw), the position of end effector height and the robot base during collaboration. Significant safety-relevant influencing factors such as speed of the moving robot parts, safety distance, vulnerable human body parts, number and duration of exposure to hazards can thus be assessed.

4. Conclusion and Future Work

Collaborative work systems in an industrial context are currently limited if changes need to be taken into account regularly. Although robot programming of modern sensitive robots is aimed for users with limited programming skills and becomes more and more sophisticated, safety regulations limit this flexibility. An advanced structured approach for safety assessment, as described in this paper, enables safe implementation of modifications to a known extent. Future work will include an extensive comparison between simulated system modifications and modifications on the real experimental setup. Furthermore, qualitative differences, e.g. in terms of perceived physical workload for the operator will also be analyzed.

Acknowledgments

The research leading to these results originate from the project DR.KORS – Dynamic reconfigurability of collaborative robot systems (FFG project no. 864892) which has received funding from the Production of the Future programme. Production of the Future is a research, technology and innovation funding programme of the Republic of Austria, Ministry of Climate Action.

We would like to thank Lukas Kaiser, Stefanie Puschl-Schliefnig, and Thomas Stähle for setting up the lab use case and performing the simulations.

References

[1] A. Djuric, J. Rickli, J. Sefcovic, D. Hutchison, and M. M. Goldin, "Integrating Collaborative Robots in Engineering and Engineering Technology Programs," *ASME International Mechanical Engineering Congress and Exposition*, vol. 5, 2018.

[2] M. Brandstötter et al., "Versatile collaborative robot applications through safety-rated modification limits," in *Advances in Service and Industrial Robotics* (K. Berns and D. Görges, eds.), (Cham), pp. 438–446, Springer International Publishing, 2020.

[3] M. J. Rosenstrauch and J. Krüger, "Safe human-robot-collaboration-introduction and experiment using iso/ts 15066," in *2017 3rd International Conference on Control, Automation and Robotics (ICCAR)*, pp. 740–744, 2017.

[4] B. Matthias, S. Kock, H. Jerregard, M. Kallman, I. Lundberg, and R. Mellander, "Safety of collaborative industrial robots: Certification possibilities for a collaborative assembly robot concept," in *2011 IEEE International Symposium on Assembly and Manufacturing (ISAM)*, pp. 1–6, 2011.

[5] M. Bdiwi, M. Pfeifer, and A. Sterzing, "A new strategy for ensuring human safety during various levels of interaction with industrial robots," *CIRP Annals*, vol. 66, no. 1, pp. 453–456, 2017.

[6] F. Vicentini, "Terminology in safety of collaborative robotics," *Robotics and Computer-Integrated Manufacturing*, vol. 63, 2020.

[7] L. Lestingi and S. Longoni, *HRC-TEAM: A Model-driven Approach to Formal Verification and Deployment of Collaborative Robotic Applications*. Project thesis, Politecnico di Milano, 2017.

[8] J. Saenz, R. Behrens, E. Schulenburg, H. Petersen, O. Gibaru, P. Neto, and N. Elkmann, "Methods for considering safety in design of robotics applications featuring human-robot collaboration," *International Journal of Advanced Manufacturing Technology*, vol. 107, p. 2313–2331, 2020.

[9] T. Ogure, Y. Nakabo, S. Jeong, and Y. Yamada, "Risk management simulator for low-powered human-collaborative industrial robots," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 49–54, 2009.

[10] J. A. Marvel, J. Falco, and I. Marstio, "Characterizing task-based human–robot collaboration safety in manufacturing," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 45, no. 2, pp. 260–275, 2015.

[11] A. M. Zanchettin, N. M. Ceriani, P. Rocco, H. Ding, and B. Matthias, "Safety in human-robot collaborative manufacturing environments: Metrics and control," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 2, pp. 882–893, 2015.

[12] T. Ritchey, "General morphological analysis," in *16th euro conference on operational analysis*, 1998.

Several Approaches for the Optimization of Arm Motions of Humanoids

Daniel Lichtenecker, Gabriel Krög, Hubert Gattringer and Andreas Müller
Institute of Robotics, Johannes Kepler University Linz,
Altenbergerstraße 69, 4040 Linz, Austria
www.robotik.jku.at
{daniel.lichtenecker,gabriel.kroeg,hubert.gattringer,a.mueller}@jku.at

Abstract. *This paper presents several point-to-point optimization tasks of humanoid arm motions. The focus lies on optimization of elementary arm motions. Several cost functions for optimization tasks are defined. Tasks in respect of time optimal control, minimizing joint loads and maximizing the vertical torque of the torso are presented. The dynamic optimal control problem is transformed into a static parametric optimization problem by using B-spline curves. The optimization is carried out with the Sequential Quadratic Programming algorithm.*

1. Introduction

In general, robotic systems as humanoids are complex structures which are able to interact with their environment. The research on humanoid robots is a major and challenging part in the field of robotics. Various humanoid robotic systems have been developed in the past, see e.g. [5, 7]. Moreover, a humanoid walking machine is developed at the Johannes Kepler University Linz [6]. Fig. 1 shows the modular setup of the humanoid by means of submodules. In this configuration, the system possesses of 6 degrees of freedom (DOFs) per leg and 1 DOF per arm.

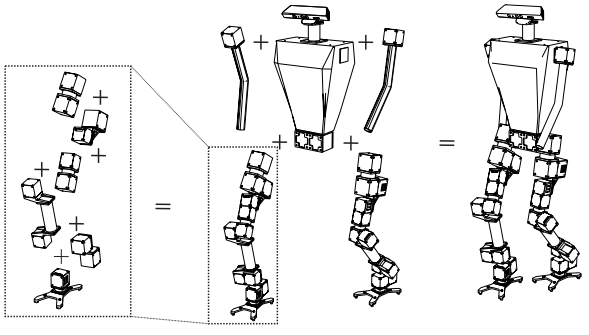


Figure 1: Schematic representation of the biped

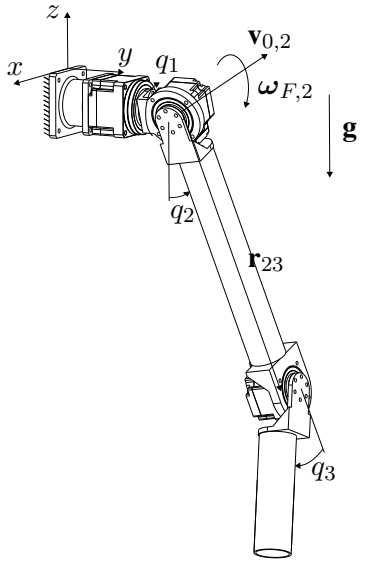


Figure 2: Schematic representation of the analyzed arm system

Especially the arm plays an important role regarding interaction or manipulation of an object. Moreover, arms are used to counterbalance the torque about the vertical axis. In order to achieve a higher degree of mobility of the arm submodule, a new arm system is developed. The system consists of 3 actuators and structural elements which are modeled as rigid bodies. The shoulder is represented by 2 DOFs and the elbow has 1 DOF. That setup is an approach toward the 7 DOFs arm module as presented in [1]. In this paper the arm module shown in Fig. 2 is considered (which will replace the rigid arm in Fig. 1). The paper focuses on optimization of elementary arm motions with respect to various goals. Start and final configurations of the arm system will be regarded as known from the human gait. As shown in Fig. 2, the arm system is spatial fixed for all further investiga-

tions.

2. Dynamic Modeling

In general, a multibody model describes the full dynamical behavior of a system. The equations of motion can be developed by the Projection Equation [2]. This method is efficient to derive the dynamic of recurrent subsystems. A typical subsystem in robotics consists of structural elements and actuators.

2.1. Subsystem Modeling

As already mentioned, modeling by means of subsystems is useful for robotic systems. Moreover, constraint forces and torques $\mathbf{Q}^c$ for coupling subsystems can be determined without additionally effort. The Projection Equation in subsystem representation is given by

$$\sum_{n=1}^{N_{sub}} \left(\frac{\partial \dot{\mathbf{y}}_n}{\partial \dot{\mathbf{q}}} \right)^\top \underbrace{\{\mathbf{M}_n \ddot{\mathbf{y}}_n + \mathbf{G}_n \dot{\mathbf{y}}_n - \mathbf{Q}_n\}}_{\mathbf{Q}_n^c} = \mathbf{0}, \quad (1)$$

$$\mathbf{Q}_n^c = \sum_{i=1}^{N_n} \left[\left(\frac{\partial \mathbf{v}_c}{\partial \dot{\mathbf{y}}_n} \right)^\top \left(\frac{\partial \boldsymbol{\omega}_c}{\partial \dot{\mathbf{y}}_n} \right)^\top \right]_i \times \left[\begin{array}{c} \dot{\mathbf{p}} + \tilde{\boldsymbol{\omega}}_R \, \mathbf{p} - \mathbf{f}^e \\ \dot{\mathbf{L}} + \tilde{\boldsymbol{\omega}}_R \, \mathbf{L} - \mathbf{M}^e \end{array} \right]_i \quad (2)$$

with N_{sub} subsystems and N_n bodies. The absolute velocity of the center of mass and the angular velocity of the i-th body are represented by $\mathbf{v}_{c,i} \in \mathbb{R}^3$ and $\boldsymbol{\omega}_{c,i} \in \mathbb{R}^3$, $\boldsymbol{\omega}_{R,i} \in \mathbb{R}^3$ is the angular velocity of a chosen body fixed reference frame. The vector of linear momentum and the vector of angular momentum are given by $\mathbf{p}_i = m_i \mathbf{v}_{c,i}$ and $\mathbf{L}_i = \mathbf{J}_i^c \boldsymbol{\omega}_{c,i}$. Mass and inertia tensor are denoted m_i and $\mathbf{J}_i^c \in \mathbb{R}^{3,3}$, respectively. Impressed forces and torques are given by $\mathbf{f}_i^e \in \mathbb{R}^3$ and $\mathbf{M}_i^e \in \mathbb{R}^3$. The vector $\mathbf{q} \in \mathbb{R}^N$ represent the N minimal coordinates of the system. The describing velocities of each subsystem are given by

$$\dot{\mathbf{y}}_n = \left(\mathbf{v}_0^\top \; \boldsymbol{\omega}_F^\top \, \dot{q} \right)_n^\top \in \mathbb{R}^7, \quad (3)$$

where $\mathbf{v}_{0,n}$ is the translational velocity of the coupling point, $\boldsymbol{\omega}_{F,n}$ is the guidance rotational velocity and $\dot{q}$ is the relative joint velocity of the n-th subsystem. In this paper, the 3 rotational coordinates $\mathbf{q} = (q_1 \; q_2 \; q_3)^\top$ are introduced as DOFs. Moreover, 3 subsystems are considered to derive the equations of motion. The describing velocities of the second subsystem can be seen in Fig. 2.

2.2. Joint Forces and Torques

As shown by 2.1, the occurring reaction forces and torques of the n-th subsystem can be determined by

$$\mathbf{Q}_n^c = \mathbf{M}_n \ddot{\mathbf{y}}_n + \mathbf{G}_n \dot{\mathbf{y}}_n - \mathbf{Q}_n, \quad (4)$$

with the mass matrix of the subsystem $\mathbf{M}_n \in \mathbb{R}^{7,7}$, the matrix of centrifugal and Coriolis forces $\mathbf{G}_n \in \mathbb{R}^{7,7}$ and the vector of forces on the subsystem $\mathbf{Q}_n \in \mathbb{R}^7$. Without projection into minimal coordinates, joint forces and torques regarding the three subsystems are given by

$$\begin{pmatrix} {}_1\mathbf{Q}^c \\ {}_2\mathbf{Q}^c \\ {}_3\mathbf{Q}^c \end{pmatrix} = \begin{bmatrix} \mathbf{E} & \mathbf{T}_{21}^\top & \mathbf{T}_{31}^\top \\ \mathbf{0} & \mathbf{E} & \mathbf{T}_{32}^\top \\ \mathbf{0} & \mathbf{0} & \mathbf{E} \end{bmatrix} \begin{pmatrix} \mathbf{Q}_1^c \\ \mathbf{Q}_2^c \\ \mathbf{Q}_3^c \end{pmatrix}. \quad (5)$$

The matrix

$$\mathbf{T}_{np} = \begin{pmatrix} \mathbf{R}_{np} & \mathbf{R}_{np} \, {}_p\tilde{\mathbf{r}}_{pn}^\top & \mathbf{R}_{np} \, {}_p\tilde{\mathbf{r}}_{pn}^\top \mathbf{e}_D \\ \mathbf{0} & \mathbf{R}_{np} & \mathbf{R}_{np} \mathbf{e}_D \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{pmatrix} \quad (6)$$

maps a quantity from the predecessor frame p into the frame of interest n. $\mathbf{R}_{np} \in \mathbb{R}^{3,3}$ is the rotation matrix to transform coordinate vectors resolved in the frame p into frame n, ${}_p\mathbf{r}_{pn} \in \mathbb{R}^3$ is the displacement vector from the coupling point at the predecessor frame p to that on frame n and the vector $\mathbf{e}_D \in \mathbb{R}^3$ is the axis of rotation. The transformation matrix is a result of the kinematical chain [4].

3. Problem Definition

This section reports on different optimization tasks for point-to-point (PTP) trajectory planning. In this paper, the optimal dynamic motion problem is transformed into a static parametric optimization problem. The joint trajectories are represented by B-spline curves parameterized by a set of control points $\mathbf{d} = (d_{1,1} \; \cdots \; d_{1,n} \, d_{2,1} \; \cdots \; d_{2,n} \, d_{3,1} \; \cdots \; d_{3,n})^\top$, i.e. $\mathbf{q} = \mathbf{q}(\mathbf{d})$ and thus $\dot{\mathbf{q}} = \dot{\mathbf{q}}(\mathbf{d})$ and $\ddot{\mathbf{q}} = \ddot{\mathbf{q}}(\mathbf{d})$. For practical applications, several physical restrictions of the robotic system have to be considered. In this paper, constraints regarding to initial and final state, minimal and maximal joint angles as well as maximal motor velocities and torques are regarded. The mathematical formulation of the constraints are given in Eq. (8)–(14). The task of trajectory optimization leads to a non-linear optimization problem (NLP). Different cost functions are presented in the following.

3.1. Time Optimal Control

The optimization problem for time optimal control is defined as

$$\min_{t_f,\mathbf{d}} \int_0^{t_f} 1dt \tag{7}$$

s.t.

$$\mathbf{q}(0,\mathbf{d}) = \mathbf{q}_0 \tag{8}$$

$$\mathbf{q}(t_f,\mathbf{d}) = \mathbf{q}_{t_f} \tag{9}$$

$$\dot{\mathbf{q}}(0,\mathbf{d}) = \mathbf{0} \tag{10}$$

$$\dot{\mathbf{q}}(t_f,\mathbf{d}) = \mathbf{0} \tag{11}$$

$$\mathbf{q}_{\text{min}} \leq \mathbf{q}(\mathbf{d}) \leq \mathbf{q}_{\text{max}} \tag{12}$$

$$-\dot{\mathbf{q}}_{\text{max}} \leq \dot{\mathbf{q}}(\mathbf{d}) \leq \dot{\mathbf{q}}_{\text{max}} \tag{13}$$

$$-\mathbf{Q}_{\text{max}} \leq \mathbf{Q}(\mathbf{d}) \leq \mathbf{Q}_{\text{max}} \tag{14}$$

$$\mathbf{Q}(\mathbf{q}(\mathbf{d})) = \mathbf{M}(\mathbf{q}(\mathbf{d}))\ddot{\mathbf{q}}(\mathbf{d}) + \mathbf{g}(\mathbf{q}(\mathbf{d}),\dot{\mathbf{q}}(\mathbf{d})). \tag{15}$$

In this case, the final time t_f and the set of control points $\mathbf{d}$ to parameterize the B-splines are regarded as optimization variables. Eq. (15) represents the dynamical behavior of the robotic system in minimal representation. $\mathbf{M} \in \mathbb{R}^{3,3}$ is the global mass matrix, $\mathbf{g} \in \mathbb{R}^3$ includes non-linear terms and $\mathbf{Q} \in \mathbb{R}^3$ is the global vector of generalized forces. The restrictions in Eq. (8)–(12) are associated to process requirements and those in Eq. (13)–(14) are defined by chosen motors. This equality and inequality constraints were used for all optimization tasks in the following.

3.2. Minimizing Joint Loads

Aim of this optimization task is to minimize dynamic joint forces and torques between ground/torso and arm of the humanoid walking machine. The final time t_f for the motion is predefined in this task. The cost function is given by

$$\min_{\mathbf{d}} {}_1\mathbf{Q}^{c^\top}\,{}_1\mathbf{Q}^c. \tag{16}$$

The set of control points $\mathbf{d}$ are regarded as optimization variables. As mentioned above, the optimization constraints are given by Eq. (8)–(14). Note, the occurring joint forces and torques can be calculated with Eq. (5).

3.3. Maximizing the Vertical Torque of the Torso

During gait, arms are used to counterbalance the torque around the vertical axis. A momentum control approach with this quantity is presented in [6]. Hence, another optimization strategy is to find a proper set of control points $\mathbf{d}$ such that the cost function

$$\max_{\mathbf{d}} {}_1Q_6^{c^\top}\,{}_1Q_6^c \tag{17}$$

is maximized. Once again, optimization constraints are given by Eq. (8)–(14). The quantity ${}_1Q_6^c$ is the sixth entry of ${}_1\mathbf{Q}^c$ and describes the joint torque of the first subsystem in the opposite direction of the gravity vector.

4. Optimization Method

The Sequential Quadratic Programming (SQP) algorithm was chosen to solve all considered optimization problems. This approach is also used in [3] for trajectory planning. The SQP method requires an valid initial guess for trajectories. In this case, the initial trajectories are defined as B-splines. Properties of B-splines can be found in [8]. An initial guess for the arm angels $\mathbf{q}$ are found by interpolating the initial and final position as well as some support points with a B-spline of degree 4. Furthermore, velocities and accelerations at the initial and final position are set to zero. Twenty control points were chosen to initialize each of the three polynomials. Fig. 3 shows exemplary an initial guess trajectory.

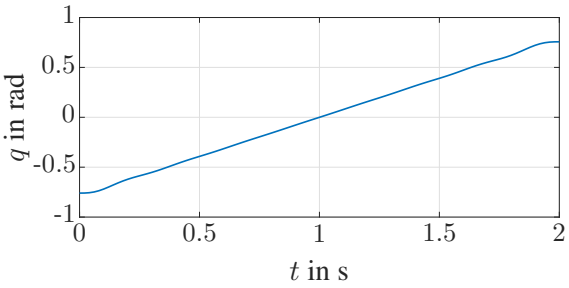


Figure 3: Example of an initial guess trajectory

5. Simulation Results

In this section, relevant results of the optimization tasks are presented. The typical arm motion during a step of the biped is defined by the start configuration $\mathbf{q}_0 = \left(-\frac{\pi}{4}\ 0\ 0\right)^\top \text{rad}$ and the final configuration $\mathbf{q}_{t_f} = \left(\frac{\pi}{4}\ 0\ \frac{\pi}{4}\right)^\top \text{rad}$ of the minimal coordinates. Moreover, limits regarding joint angles are defined by $\mathbf{q}_{\text{min}} = \left(-\frac{\pi}{2}\ 0\ 0\right)^\top \text{rad}$ and $\mathbf{q}_{\text{max}} = \left(\pi\ \pi\ \frac{3\pi}{4}\right)^\top \text{rad}$. Maximal motor rotational velocities and torques are given by $\dot{\mathbf{q}}_{\text{max}} = (12.6\ 6.3\ 12.6)^\top \text{rads}^{-1}$ and $\mathbf{Q}_{\text{max}} = (415\ 480\ 165)^\top \text{N m}$. Note, that all motor torques

and rotational velocities of the following figures are normalized w.r.t. their physical limits.

5.1. Time Optimal Control

Fig. 4 shows the normalized motor torques and rotational velocities of time optimal control problem. For time optimal optimization tasks, it is obvious that at least one motor restriction is active. The final time is given by $t_f = 0.83\,\mathrm{s}$. Moreover, all physical limits of the motors are well considered. Occurring joint forces and torques of the first subsystem are shown in Fig. 5.

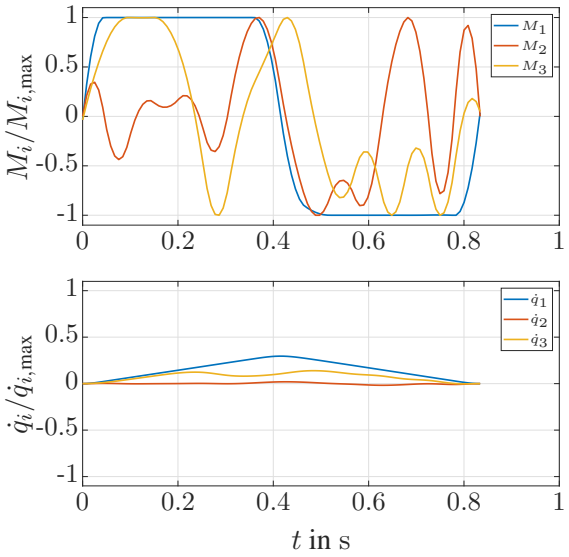


Figure 4: Normalized motor torques and rotational velocities of the time optimal control

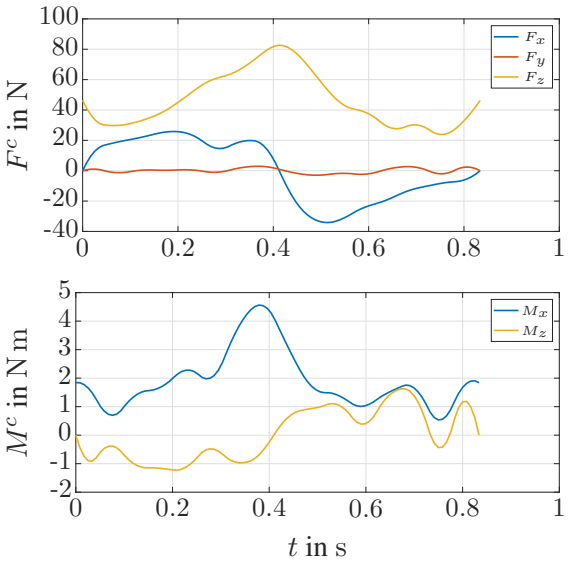


Figure 5: Joint forces and torques of the time optimal control

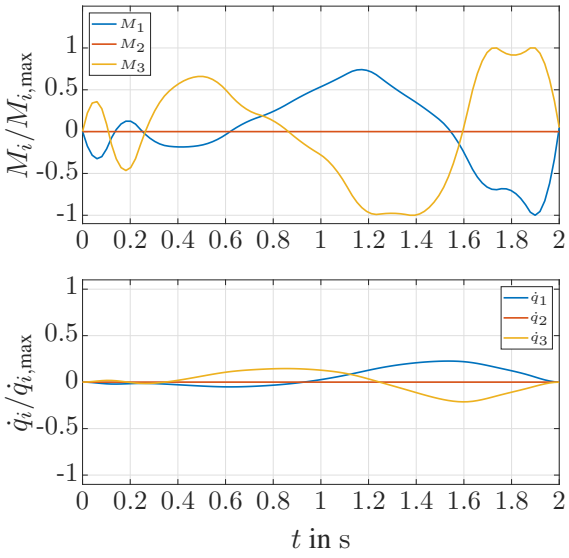


Figure 6: Normalized motor torques and rotational velocities of the joint load minimization

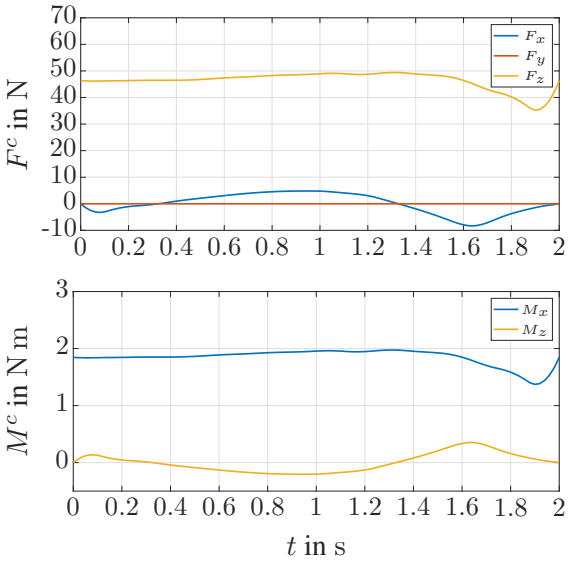


Figure 7: Joint forces and torques of the joint load minimization

5.2. Minimizing Joint Loads

In comparison to time optimal control, the final time $t_f = 2\,\mathrm{s}$ of this optimal control is predefined due to the walking speed of the analyzed biped. Fig. 6 and Fig. 7 shows the dynamical behavior of this task. Motor restrictions are almost inactive and the occurring joint forces and torques are reduced in comparison to Fig. 5.

5.3. Maximizing the Vertical Torque of the Torso

As to the last subsection 5.2, the final time $t_f = 2\,\mathrm{s}$ is also predefined in this task. Optimization re-

sults of the maximization task are shown in the following figures. Almost all motor torque restrictions are active due to the maximization. As can be seen in Fig. 8, the first arm moves at the start in the negative direction. The resulting motion performs as a swing-up procedure.

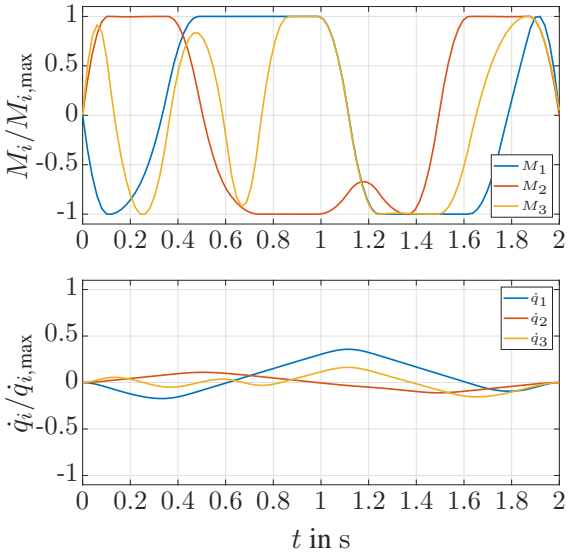


Figure 8: Normalized motor torques and rotational velocities of the vertical torque maximization

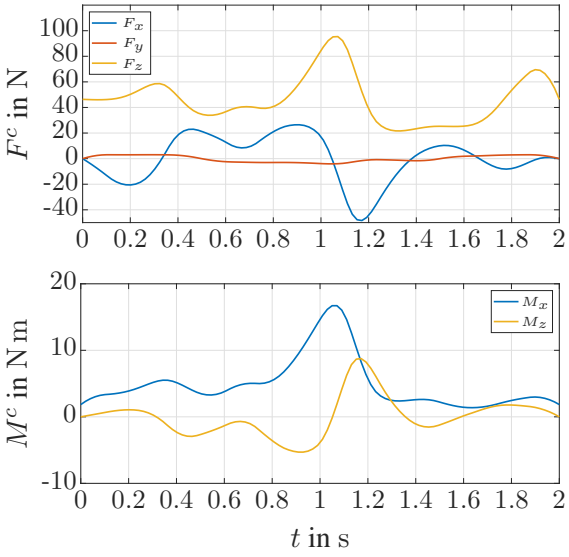


Figure 9: Joint forces and torques of the vertical torque maximization

6. Conclusion

Arm systems of humanoids are used in different ways, e.g. to counterbalance the torque around the vertical axis. Motion planning in relation to various tasks becomes an important role. Therefore, in this paper different optimization goals regarding arm motions were analyzed. Cost functions with respect to time optimal control, joint load minimization and vertical torque maximization were considered. The dynamic optimization process has been converted into a static optimization process by using B-splines curves to formulate trajectories. The NLP were solved with the SQP method.

Acknowledgments

This work has been partially supported by the "LCM – K2 Center for Symbiotic Mechatronics" within the framework of the Austrian COMET-K2 program. Additional support is provided by the Linz Institute of Technology.

References

[1] M. Benati, S. Gaglio, P. Morasso, V. Tagliasco, and R. Zaccaria. Anthropomorphic robotics. *I. Representing Mechanical Complexity*, pages 125–140, 180.

[2] H. Bremer. *Elastic Multibody Dynamics: A Direct Ritz Approach*. Springer Verlag, 2008.

[3] T. Chettibi, H. Lehtihet, M. Haddad, and S. Hanchi. Minimum cost trajectory planning for industrial robots. *European Journal of Mechanics - A/Solids*, 23(4):703 – 715, 2004.

[4] H. Gattringer. *Starr-elastische Robotersysteme: Theorie und Anwendungen*. Springer Verlag, 2011.

[5] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka. The development of honda humanoid robot. In *Proceedings. 1998 IEEE International Conference on Robotics and Automation*, volume 2, pages 1321–1326, 1998.

[6] J. Mayr. *Development and Control of a Modular Bipedal Walking Robot*. Trauner Verlag, 2016.

[7] G. Nelson, A. Saunders, N. Neville, B. Swilling, J. Bondaryk, D. Billings, C. Lee, R. Playter, and M. Raibert. Petman: A humanoid robot for testing chemical protective clothing. *Journal of the Robotics Society of Japan*, 30(4):372–377, 2012.

[8] L. Piegl and W. Tiller. *The NURBS Book*. Springer Verlag, 1997.

OAGM Workshop

Presentation Attacks and Detection in Finger- and Hand-Vein Recognition

Luca Debiasi, Christof Kauba, Heinz Hofbauer, Bernhard Prommegger, Andreas Uhl
Department of Computer Sciences, University of Salzburg
{ldebiasi,ckauba,hhofbaue,bprommeg,uhl}@cs.sbg.ac.at

Abstract. *Biometric recognition systems, especially vascular pattern based ones, are becoming more popular. However, these systems are still susceptible to so called presentation attacks, where a forged representation of the original biometric is presented to the system trying to mimic the original biometric and fool the system. We propose a presentation attack approach for finger- and hand-vein recognition systems using paper prints as well as wax and silicone artefacts. We further develop a suitable presentation attack detection (PAD) scheme based on natural scene statistics and acquire a corresponding hand vein presentation attack dataset. Evaluating the PAD scheme on the dataset confirmed its success in the detection of the forged samples.*

1. Introduction

In our modern world there is an ever growing need for personal authentication. Biometric authentication systems are one way to overcome the typical problems of classical authentication methods, e.g. disclosed or forgotten passwords, lost or stolen keys and forged signatures. Biometric authentication systems are based on so called biometric traits, which are unique behavioural or physiological characteristics of a person. These are inherently linked to a person and cannot get lost, be forgotten or be stolen. The most prominent examples of biometric traits include fingerprints, face and iris. Recently, vascular pattern based biometrics (usually denoted as vein recognition based systems) gain more attention as well, with finger- and hand-vein based systems being the most widely used ones [27]. Vein based systems exhibit some advantages over other biometric systems, e.g. fingerprint and face recognition ones. They rely on the structure of the vascular pattern formed by the blood vessels inside the human body tissue, i.e. it is an internal biometric trait. This pattern only becomes visible in near-infrared (NIR) light, as the haemoglobin in the blood absorbs NIR light, rendering the blood vessels (veins) visible as dark lines in the captured images. Vein based systems are more resistant to forgery and they are neither susceptible to abrasion nor skin surface conditions [11].

Despite the advantages mentioned above, biometric recognition systems are far from being perfect. Almost all of the currently employed systems are susceptible to spoofing or presentation attacks (PAs). A PA is defined as *presentation to the biometric data capture subsystem with the goal of interfering with the operation of the biometric system* according to the ISO/IEC 30107-1 standard [4]. This corresponds to the creation of a forged representation mimicking the original biometric trait (also called a spoofing artefact) that is used to spoof/fool the biometric system. PAs are posing a severe problem in practical applications as a genuine user may be impersonated. By launching a successful PA, an adversary is able to gain illegitimate access to the system. In contrast to passwords and tokens, a biometric trait can neither be replaced nor revoked. Hence, if a system is prone to PAs, it can no longer be considered as secure. Fortunately, there are counter-measures which aim to detect PAs by equipping the biometric system with either additional hardware or software performing presentation attack detection (PAD).

In this work we focus on PAs and PAD for finger- as well as hand-vein recognition systems. We propose several approaches to create spoofing artefacts using different materials replicating the vein pattern of genuine subjects. Furthermore, corresponding PA datasets are acquired and a PAD approach, tested on hand veins, is presented.

The rest of the paper is organised as follows: Section 2 gives an overview on PAs and PAD schemes for finger- and hand-vein recognition. In Section 3 the generation of the spoofing artefacts is explained.

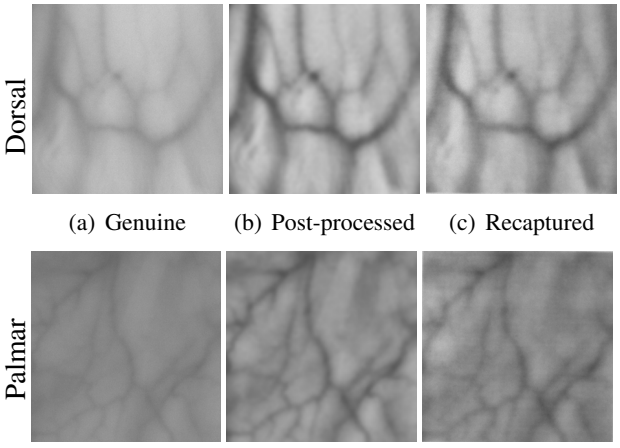
Section 4 presents our proposed PAD approach. The experimental evaluation is described in Section 5. Section 6 concludes this paper and gives an outlook on future work.

2. Related Work

Finger- and hand-veins have been shown to be susceptible to spoofing [26, 24]. PAD approaches help in detecting presentation attacks and can be categorised into liveness-based (rely on signs of vitality, e.g. capturing the heartbeat), motion-based (analyse movements during the capturing process and try to detect unnatural ones) and texture-based methods (detect and analyse textural artefacts present in the image). While the first two categories require a video or a sequence of consecutive images to be captured, texture-based methods can be applied to single images. One liveness based approach is presented in [19], which applies motion magnification techniques. The majority of the proposed PAD schemes are texture-based ones, e.g. a Fourier, Haar and Daubechies wavelet transform based one [16], exploiting differences in the bandwidth of vertical energy signals. A binarised statistical image features based one and some others based on Riesz transform, local binary patterns (LBP), local phase quantisation and Weber local descriptors are presented in [25]. Another approach [23] uses a windowed dynamic mode decomposition (W-DMD) to detect spoofed finger vein images. Even baseline LBP [20] and some LBP variants and extensions of LBP [10] proved to be effective for the task of finger vein PAD. Several other approaches are utilising image quality assessment methods (IQA), e.g. [15] and [1] which detection accuracy turns out to be highly dependent on the particular dataset. In [22] the authors showed that the classification accuracy can be improved by incorporating natural scene statistics (NSS) [13]. A very different approach for PAD detection is to use a photo-response non-uniformity (PRNU) technique to differentiate PA data from genuine samples [12]. Furthermore, a CNN-based approach has been proposed in [17].

3. Presentation Attack Approaches

Capturing the vein pattern using an appropriate capturing device forms the basis of vein recognition in general and finger- and hand-vein PA evaluation in particular. Therefore, we utilise the PLUSVein finger vein scanner [7] and the PLUS hand vein scanner [8] as capturing devices to prepare our finger- and hand-vein spoofing artefacts as well as for recapturing the artefacts. The interested reader is referred to the authors original publications for more details about those capturing devices. In the following, the production of the hand and finger vein spoofing artefacts is described. These spoofing artefacts are then again presented to the capturing devices mentioned above.



(a) Genuine (b) Post-processed (c) Recaptured

(a) Genuine (b) Post-processed (c) Recaptured

Figure 1. Hand vein PA artefacts for 950 nm reflected light illumination captured with the PLUS hand vein scanner [8]: genuine image (a), post-processed image for printing (b) and re-acquired printed image (c).

3.1. Hand Vein Spoofing Artefacts

The hand vein capturing device is used to acquire reflected light images in two different wavelengths (850 and 950 nm). Since printouts of finger vein patterns have shown to yield successful presentation attacks [26], we decided for this approach as an attack scenario for the hand vein recognition system as well. Our spoofing attack samples are derived from samples contained in the publicly available PROTECTVein dataset, which is part of the PROTECT Multimodal Biometric Database [21].

The hand vein spoofing attack samples are created by first selecting 100 images based on the visibility of the vein pattern (5 dorsal and 5 palmar for one hand of 10 users). Afterwards, a region of interest (ROI) is manually cropped from the images. These ROIs are then post-processed using a Contrast Limited Adaptive Histogram Equalisation (CLAHE) and Gauss filtering, to enhance the visibility of the vascular pattern and remove the skin texture and hair to eventually obtain smooth images. Afterwards, the post-processed images are scaled to approximately match

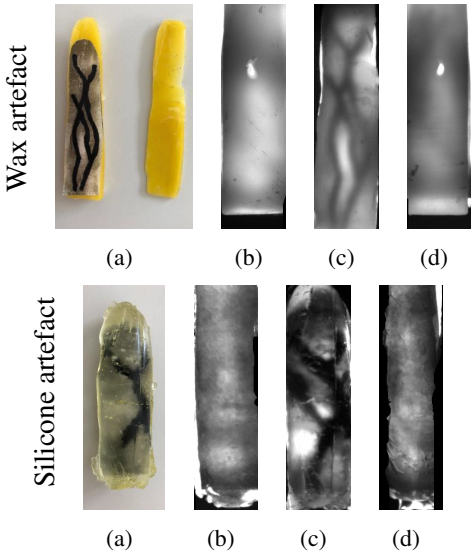


Figure 2. Wax and silicone artefacts (a) and image as captured by the PLUSVein finger vein scanner [7] using different enhancements for the vein pattern: no enhancement (b), tracing with black marker (c), local contrast enhancement (CLAHE) (d).

the real-life genuine samples and printed to paper. Multiple printers and print configurations have been tested to find an appropriate solution in regard to the absorption of NIR light. In the end, using a 'HP LaserJet 500 colour M551' laser printer in grey-scale printing mode yielded satisfactory results. Some examples of the hand vein PA artefact generation and recapturing are shown in Figure 1.

3.2. Finger Vein Spoofing Artefacts

For the light transmission based finger vein modality, the establishment of working PA artefacts is less trivial than in the reflected light case seen for hand veins. Following an idea as exhibited in a recent Chaos Computer Club video based on a sliced wax artefact and a silicone model as proposed in [18] we finally came up with two different types of artefacts, as shown in Figure 2. These artefacts are derived from samples contained in the publicly available PLUSVein-FV3 finger vein data set [6]. These two materials exhibited the best properties in regard to appropriate illumination in the light transmission case among several other considered materials.

For both types of artefacts, wax and silicone, the first step in creating the artefacts is to obtain a mould with a finger-like shape. We use a 3D-printer to create the moulds, consisting of two parts: base and top. Afterwards the vein pattern is printed using a 'HP LaserJet 500 colour M551' laser printer in grey-scale printing mode (similar to hand vein artefacts). The paper sheet containing the vein pattern is placed between the bottom and top finger artefact parts, as shown in Figure 2. The same finger artefact could be used for all spoofs by simply substituting the piece of paper containing the vein pattern.

In order to improve the visibility of the vein pattern, different techniques are employed: no enhancement, enhancing the image (CLAHE and Gauss filtering) as well as tracing the veins with a black permanent marker. Furthermore, various types of paper are tested. The tracing of the vein pattern yields the visually most pleasing results. In total, 42 finger artefacts (2 materials, 7 types of paper, 3 vein pattern enhancements) are generated for 3 fingers of an exemplary user. Figure 2 illustrates the created artefacts and images recaptured with the sensor.

4. Presentation Attack Detection

The PAD system applied in this work uses natural scene statistics as described in [13] and is based on the framework presented in [2], which was adapted to presentation attack detection in [22]. In brief, the features used for detection are the parameters of (asymmetrical) generalised Gaussian distributions, (A)GGD, fit to statistics of characteristics derived from samples & artefacts using a multi-scale approach.

The features are fed into a support vector machine (SVM) for classification, two-class 'genuine' and 'spoofed', using a radial basis function. First of all, the available genuine and spoofed data is randomly separated on a user basis into two equally sized training and test sets.

For training, in order to cleanly separate training and evaluation data, learning is done using a 'leave one label out' cross-fold technique. All images of a user's hand are defined as having the same label, i.e. the right and left hand have different labels for each user. Furthermore, also the perspective (dorsal or palmar) is split into different labels. To evaluate on the whole training dataset each label is left out in turn, the SVM is trained on the relevant training data, then the left-out label is evaluated. The final training evaluation data is the union of the individually evaluated labels. The parameters are optimised for the overall training database, where the search is done non-exhaustively on a grid with logarithmic drill-down, presenting closed set learning. The spoofing detection accuracy serves as learning

function for the parameter optimisation.

The trained SVM is then applied to the previously unseen test data and yields an output class and a confidence, which represents the difference between class probabilities.

5. Experimental Evaluation

This section describes the experimental set-up for the evaluation of the hand- (HV) and finger-vein (FV) spoofing artefacts as well as the spoofing artefact's quality and PAD performance.

5.1. Experimental Set-Up

The software used to process the finger- and hand-vein data is the OpenVein Toolkit [9]. The ROI extraction has been done manually and the visibility of the vein pattern is improved by applying different post-processing techniques from the toolkit. The vascular patterns are extracted using Maximum Curvature (MC) [14] and the comparison of the resulting binary feature vectors is performed using a correlation based approach [14].

As defined in ISO/IEC 19795-1 [3], the EER, FMR1000 and ZeroFMR are used to quantify the verification performance, where all samples are compared against each other (full comparison). The experiments are performed separately for fingers/hands, orientations (dorsal/palmar) and illumination types where applicable.

The PAD approach is evaluated using the metrics defined in the ISO/IEC 30107-3 [5] standard: detection equal error rate (D-EER), where APCER=BPCER, attack presentation classification error rate (APCER, equivalent of FAR) which is the proportion of attack presentations using the same spoofing artefact species incorrectly classified as bona fide (true) presentations in a specific scenario, bona fide presentation classification error rate (BPCER, equivalent of FRR) representing the proportion of bona fide presentations incorrectly classified as presentation attacks in a specific scenario and a corresponding Detection Error Trade-off (DET) curve.

5.2. Results: Quality of Spoofing Artefacts

In order to assess the PAD performance, it is essential to evaluate the quality of the spoofed artefacts first. This is done by comparing the recaptured images of the spoofed artefacts against bona fide images. The main goal in creating the spoofed artefacts is to have as little as possible impact on the matching performance. If that is the case, the quality of the spoofed artefacts can be considered as satisfactory.

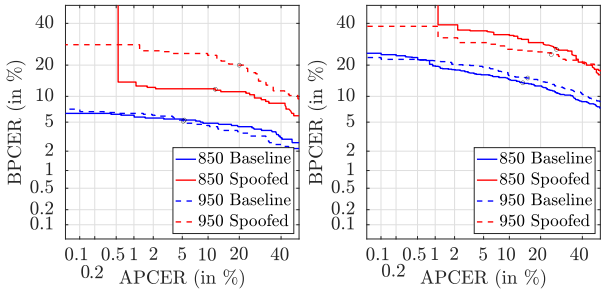


Figure 3. HV Verification results obtained when comparing bona fide samples only (baseline) and with presentation attacks (spoofed) for dorsal (left) and palmar (right) view.

		EER	FMR1000	ZeroFMR
Baseline	Dorsal 850	3.01	3.00	4.00
	Dorsal 950	4.99	6.00	6.00
	Palmar 850	16.99	30.00	32.00
	Palmar 950	18.16	32.00	33.00
Spoofed	Dorsal 850	10.80	94.80	98.00
	Dorsal 950	11.20	15.60	16.40
	Palmar 850	20.82	100.00	100.00
	Palmar 950	23.22	38.00	41.20

Table 1. Performance values (in %) obtained when verifying bona fide samples only (baseline) compared to verifying bona fide samples against PAs (spoofed) for reflected light HV recognition.

The results for the HV artefacts (reflected light) are shown in Figure 3 and the corresponding performance values are reported in Table 1. In general, we notice a matching performance degradation with spoofing artefacts, however the resulting EER degradation is still acceptable. It can be observed that the quality of the 950 artefacts (dorsal and palmar) is consistent for all spoofed patterns, since the FMR1000 and ZeroFMR remain quite stable in this case. For the 850 spoofs on the other hand, a large degradation in the FMR1000 and ZeroFMR can be observed, which indicates that some of the created artefacts did not have sufficient quality. Furthermore, the baseline performance is much lower for the palmar view compared to the dorsal one (3.01% vs. 16.99%), while the relative EER degradation using spoofed artefacts behaves stably and ranges approximately between 4% and 7% for all modalities.

Table 2 illustrates the comparison scores (genuine and impostor) of the created FV spoofing artefacts compared to the baseline, where only bona fide

Artefact Type	aGen	aImp
Baseline	0.2346	0.1257
Wax	0.1222	0.1236
Wax traced	0.1199	0.1220
Wax CLAHE	0.1252	0.1199
Silicone	0.1285	0.1297
Silicone traced	0.1250	0.1250
Silicone CLAHE	0.1274	0.1283

Table 2. Average genuine (aGen) and impostor (aImp) FV comparison scores obtained when verifying bona fide samples only (baseline) compared to verifying bona fide samples against PAs using different artefact types for light transmission FV recognition.

images have been used. The scores have been averaged over all three fingers and paper types for illustration purposes because of the small variation in their scores. It is immediately noticeable that none of the spoofing artefacts is meeting the quality requirements, since the obtained genuine and impostor scores are not differentiable. This is also true for the visually promising traced wax artefacts. Therefore, a further refinement of these artefacts is necessary to come up with a dataset of sufficient quality as required for a sensible PAD evaluation.

5.3. Results: PAD Performance

Following the evaluation of the produced spoofing artefacts' quality, this section covers the detection performance of the PAD system described in section 4. The evaluation of the PAD system is only performed for presentation attacks using HV artefacts due to insufficient quality of the FV artefacts. The available genuine and spoofed data was split 50/50 on a user basis for training and testing.

The PAD detection performance under different illumination conditions in terms of D-EER, BPCER @ APCER<=0.001 (BPCER1000) and BPCER @ APCER=0 (BPCER0) is reported in Table 3. It becomes apparent that the artefacts are harder to detect under 950nm NIR than under 850nm one. This might be due to varying reflectivity and absorption properties of the vein pattern prints for different NIR wavelengths. The PAD system has some problems in correctly classifying the palmar 950nm artefacts, however the PAD performance can be considered good to excellent across all HV artefacts.

	D-EER	BPCER1000	BPCER0
Dorsal 850	0.22	0.43	0.43
Dorsal 950	0.33	0.65	0.65
Palmar 850	0.00	0.00	0.00
Palmar 950	6.04	30.43	30.43

Table 3. Performance values (in %) for hand veins PAD evaluation

6. Conclusion

Presentation attacks are still a major problem in many applications of biometric recognition systems. Recent publications have shown that even vascular pattern based systems are susceptible to this kind of attack. In this work, we investigated two approaches to produce presentation attack artefacts, one for finger veins and one for hand veins. We also developed a suitable presentation attack detection scheme for vein recognition systems based on a natural scene statistics framework. We established a hand vein presentation attack dataset, consisting of 100 presentation attack samples and the corresponding original samples, which is publicly available as part of the PROTECT MMDB v2[1].

The PAD evaluation results on the established dataset showed that the proposed PAD approach achieves a good performance in detecting the fake representations. The verification experiment further revealed that if the fake representations are not detected, they achieve a rather high verification rate, i.e. that there is a good chance that a presentation attack is successful if no suitable PAD approach is employed.

Our future work will include tests with other types of presentation attack artefacts for the hand veins and the establishing of a presentation attack dataset for finger veins as well.

References

[1] A. P. S. Bhogal, D. Söllinger, P. Trung, J. Hämmerle-Uhl, and A. Uhl. Non-reference image quality assessment for fingervein presentation attack detection. In *Scandinavian Conference on Image Analysis*. Springer, 2017.

[2] H. Hofbauer and A. Uhl. Applicability of no-reference visual quality indices for visual security assessment. In *Proceedings of the 6th ACM Workshop on Information Hiding and Multimedia Security*, 2018.

[1] Will be released at `http://projectprotect.eu`

[3] ISO/IEC JTC1 SC37 Biometrics. ISO/IEC IS 19795-1:2006, it – Biometric performance testing and reporting – Part 1: Principles and framework.

[4] ISO/IEC JTC1 SC37 Biometrics. *ISO/IEC IS 30107-1:2016, IT – Biometric presentation attack detection – Part 1: Framework*.

[5] ISO/IEC JTC1 SC37 Biometrics. *ISO/IEC IS 30107-3:2017, IT – Biometric presentation attack detection – Part 3: Testing and Reporting*.

[6] C. Kauba, B. Prommegger, and A. Uhl. Focussing the beam - a new laser illumination based data set providing insights to finger-vein recognition. In *2018 IEEE 9th Int. Conference on Biometrics Theory, Applications and Systems (BTAS)*, Los Angeles, California, USA, 2018.

[7] C. Kauba, B. Prommegger, and A. Uhl. Openvein - an open-source modular multipurpose finger vein scanner design. In A. Uhl, C. Busch, S. Marcel, and R. Veldhuis, editors, *Handbook of Vascular Biometrics*, chapter 3. Springer Nature Switzerland AG, Cham, Switzerland, 2019.

[8] C. Kauba and A. Uhl. Shedding light on the veins - reflected light or transillumination in hand-vein recognition. In *Proceedings of the 11th IAPR/IEEE Int. Conference on Biometrics (ICB'18)*, Gold Coast, Queensland, Australia, 2018.

[9] C. Kauba and A. Uhl. An available open-source vein recognition framework. In A. Uhl, C. Busch, S. Marcel, and R. Veldhuis, editors, *Handbook of Vascular Biometrics*, chapter 4. Springer Nature Switzerland AG, Cham, Switzerland, 2019.

[10] D. Kocher, S. Schwarz, and A. Uhl. Empirical evaluation of lbp-extension features for finger vein spoofing detection. In *2016 Int. Conference of the Biometrics Special Interest Group (BIOSIG)*. IEEE, 2016.

[11] A. Kumar and Y. Zhou. Human identification using finger images. *Image Processing, IEEE Transactions on*, 21(4), 2012.

[12] B. Maser, D. Söllinger, and A. Uhl. Prnu-based detection of finger vein presentation attacks. In *2019 7th Int. Workshop on Biometrics and Forensics (IWBF)*, 2019.

[13] A. Mittal, A. K. Moorthy, and A. C. Bovik. No-reference image quality assessment in the spatial domain. *IEEE Transactions on image processing*, 21(12), 2012.

[14] N. Miura, A. Nagasaka, and T. Miyatake. Extraction of finger-vein patterns using maximum curvature points in image profiles. *IEICE transactions on information and systems*, 90(8), 2007.

[15] B. Mythily and K. Sathyaseelan. Measuring the quality of image for fake biometric detection: application to finger vein. In *National conference on research advances in communication, computation, electrical science and structures (NCRACCESS)*, 2015.

[16] D. T. Nguyen, Y. H. Park, K. Y. Shin, S. Y. Kwon, H. C. Lee, and K. R. Park. Fake finger-vein image detection based on fourier and wavelet transforms. *Digital Signal Processing*, 23(5), 2013.

[17] D. T. Nguyen, H. S. Yoon, T. D. Pham, and K. R. Park. Spoof detection for finger-vein recognition system using nir camera. *Sensors*, 17(10), 2017.

[18] X. Qiu, W. Kang, S. Tian, W. Jia, and Z. Huang. Finger vein presentation attack detection using total variation decomposition. *IEEE Transactions on Information Forensics and Security*, 13(2), 2017.

[19] R. Raghavendra, M. Avinash, S. Marcel, and C. Busch. Finger vein liveness detection using motion magnification. In *2015 IEEE 7th Int. Conference on Biometrics Theory, Applications and Systems (BTAS)*. IEEE, 2015.

[20] R. Raghavendra and C. Busch. Presentation attack detection algorithms for finger vein biometrics: A comprehensive study. In *2015 11th Int. Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*. IEEE, 2015.

[21] A. F. Sequeira, J. Ferryman, L. Chen, C. Galdi, J.-L. Dugelay, V. Chiesa, A. Uhl, B. Prommegger, C. Kauba, S. Kirchgasser, A. Grudzien, M. Kowalski, L. Szklarski, P. Maik, and P. Gmitrowicz. PROTECT Multimodal DB: a multimodal biometrics dataset envisaging border control. In *Proceedings of the Int. Conference of the Biometrics Special Interest Group (BIOSIG'18)*, Darmstadt, Germany, 2018.

[22] D. Söllinger, P. Trung, and A. Uhl. Non-reference image quality assessment and natural scene statistics to counter biometric sensor spoofing. *IET Biometrics*, 7(4), 2018.

[23] S. Tirunagari, N. Poh, M. Bober, and D. Windridge. Windowed dmd as a microtexture descriptor for finger vein counter-spoofing in biometrics. In *2015 IEEE Int. Workshop on Information Forensics and Security (WIFS)*. IEEE, 2015.

[24] P. Tome and S. Marcel. On the vulnerability of palm vein recognition to spoofing attacks. In *The 8th IAPR Int. Conference on Biometrics (ICB)*, May 2015.

[25] P. Tome, R. Raghavendra, C. Busch, S. Tirunagari, N. Poh, B. Shekar, D. Gragnaniello, C. Sansone, L. Verdoliva, and S. Marcel. The 1st competition on counter measures to finger vein spoofing attacks. In *2015 Int. Conference on Biometrics (ICB)*. IEEE, 2015.

[26] P. Tome, M. Vanoni, and S. Marcel. On the vulnerability of finger vein recognition to spoofing. In *IEEE Int. Conference of the Biometrics Special Interest Group (BIOSIG)*, Sept. 2014.

[27] A. Uhl, C. Busch, S. Marcel, and R. Veldhuis. *Handbook of vascular biometrics*. Springer, 2020.

HPS: Holistic End-to-End Panoptic Segmentation Network with Interrelations

Günther Kniewasser, Alexander Grabner, Peter M. Roth
Institute of Computer Graphics and Vision, Graz University of Technology, Austria
{guenther.kniewasser@student,alexander.grabner@icg, pmroth@icg}.tugraz.at

Abstract. *To provide a complete 2D scene segmentation, panoptic segmentation unifies the tasks of semantic and instance segmentation. For this purpose, existing approaches independently address semantic and instance segmentation and merge their outputs in a heuristic fashion. However, this simple fusion has two limitations in practice. First, the system is not optimized for the final objective in an end-to-end manner. Second, the mutual information between the semantic and instance segmentation tasks is not fully exploited. To overcome these limitations, we present a novel end-to-end trainable architecture that generates a full pixel-wise image labeling with resolved instance information. Additionally, we introduce interrelations between the two subtasks by providing instance segmentation predictions as feature input to our semantic segmentation branch. This inter-task link eases the semantic segmentation task and increases the overall panoptic performance by providing segmentation priors. We evaluate our method on the challenging Cityscapes dataset and show significant improvements compared to previous panoptic segmentation architectures.*

1. Introduction

Panoptic segmentation [12] addresses the problem of complete 2D scene segmentation by not only assigning a class label to each pixel of an image but also differentiating between instances within a common class. Thus, it can be seen as a unification of semantic segmentation [22, 24, 3] and instance segmentation [8, 13, 20, 16]. Panoptic segmentation is a new and active research area with applications in augmented reality, robotics, and medical imaging [5, 23, 30].

To predict a panoptic segmentation of an image, recent approaches perform three tasks. First, they perform semantic segmentation to identify regions of uncountable *stuff* classes like sky. Second, they perform instance segmentation to detect individual instances of countable *things* classes like cars. Third, they merge the outputs of these two tasks into a single panoptic prediction.

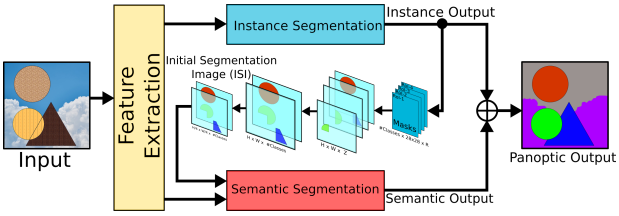


Figure 1: Illustration of our proposed panoptic segmentation network with task interrelations. We provide instance segmentation predictions as additional feature input to our semantic segmentation branch. In this way, we exploit a segmentation prior which increases the overall panoptic performance.

However, this strategy has two limitations in practice. First, because the panoptic output is generated using heuristics, the system cannot be optimized for the final objective in an end-to-end manner. Second, semantic and instance segmentation share mutual information and similarities but the relation between the two tasks is not exploited because they are addressed independently.

To overcome these limitations, we propose a *holistic* end-to-end trainable network for *panoptic segmentation* (HPS) with interrelations between the semantic and the instance segmentation branches, as shown in Figure 1. Our network directly generates a full pixel-wise image labeling with resolved instance information by using differentiable operations instead of heuristics to combine individual results. Moreover, to take advantage of mutual information between the semantic and the instance segmentation

tasks, we provide instance segmentation predictions as additional feature input to our semantic segmentation branch. In particular, we gather predicted instance masks into an *initial segmentation image* (ISI) which represents a coarse semantic segmentation for *things* classes. In this way, we exploit a segmentation prior which increases the overall panoptic performance of our system by leveraging similarities between the two previously disjoint subtasks.

We evaluate our method on the challenging Cityscapes dataset [4] for semantic understanding of urban street scenes using the recently introduced panoptic quality [11] metric. We provide an unbiased evaluation and compare four different approaches with an increasing level of entanglement between semantic and instance segmentation. Our experiments show that both end-to-end training and inter-task relations improve panoptic performance in practice.

2. Related Work

Fusing semantic and instance information has a rich history in computer vision [25, 26]. However, only recently [12] formalized the task of panoptic segmentation and introduced a panoptic quality (PQ) metric to assess the performance of complete 2D scene segmentation in an interpretable and unified manner. This formalization and the availability of large datasets with corresponding annotations [19] motivated research on panoptic segmentation.

Early approaches to panoptic segmentation use two highly specialized networks for semantic segmentation [22, 24, 3] and instance segmentation [21, 8, 17, 27] and combine their predictions heuristically [1]. Instead, recent methods address the two segmentation tasks with a single network by training a multi-task system that performs semantic and instance segmentation on top of a shared feature representation [11]. This reduces the number of parameters, the computational complexity, and the time required for training. To improve the panoptic quality, newer approaches propose a differentiable fusion of semantic and instance segmentation instead of a heuristic combination. In this way, they learn to combine the individual predictions and optimize directly for the final objective in an end-to-end manner. For example, UPSNet [28] introduces a parameter-free merging technique to generate panoptic predictions using a single network.

Another strategy to improve accuracy is to exploit mutual information and similarities between semantic and instance segmentation network branches. In this context, AUNet [15] incorporates region proposal information as an attention mechanism in the semantic segmentation branch. In this way, the semantic segmentation focuses more on *stuff* classes and less on *things* classes, which are eventually replaced by predicted instance masks. TASCNet [14] enforces L2-consistency between predicted semantic and instance segmentation masks to exploit mutual information. SOGNet [29] addresses the overlapping issue of instances using a scene graph representation which computes a relational embedding for each object based on geometry and appearance.

Similar to our approach, IMP [6] which has been developed at the same time uses predicted instance segmentation masks as additional input for the semantic segmentation branch. Compared to our approach, a different normalization technique is used and the instance masks are combined using the max operator instead of averaging.

3. Holistic End-to-End Panoptic Segmentation Network with Interrelations

An overview of our end-to-end trainable panoptic segmentation network with inter-task relations is shown in Figure 1. We first present our end-to-end trainable architecture which combines semantic and instance segmentation predictions in a differentiable way in Sec. 3.1. Then, we introduce our interrelations module which provides instance segmentation predictions as additional feature input to our semantic segmentation branch in Sec. 3.2.

3.1. End-to-End Panoptic Architecture

Our network architecture builds upon Panoptic Feature Pyramid Networks [11]. Like many recent panoptic segmentation methods, this approach extends the generalized Mask R-CNN framework [8] with a semantic segmentation branch. This results in a multi-task network that predicts a dense semantic segmentation in addition to sparse instance segmentation masks. For our implementation, we use a shared ResNet-101 [9] feature extraction backbone with a Feature Pyramid Network [18] architecture to obtain combined low- and high-level features. These features serve as shared input to our semantic and instance segmentation branches, as shown in Figure 2.

For the semantic segmentation branch, we process each stage of the feature pyramid $\{P2, \ldots, P5\}$ by a series of upsampling modules. These modules con-

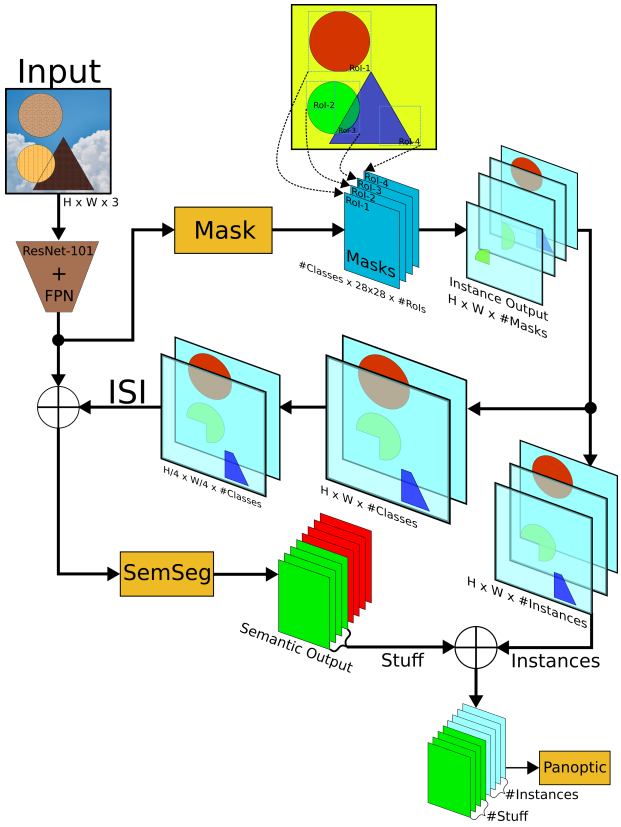


Figure 2: Detailed illustration of our end-to-end panoptic segmentation network with task interrelations. We internally merge predictions from our semantic and instance segmentation branches in a differentiable way. In particular, we concatenate *stuff* class predictions from our semantic segmentation branch with *things* class predictions in the form of canvas collections from our instance segmentation branch. Our instance canvas collections can also be transformed into an initial segmentation image (ISI) which serves as additional feature input for our semantic segmentation branch.

sists of 3×3 convolutions, batch normalization [10], ReLU [7], and $2\times$ bilinear upsampling. Because the individual stages have different spatial dimensions, we process each stage by a different number of upsampling modules to generate $H/4 \times W/4 \times 128$ feature maps, where H and W are the input image dimensions. The resulting outputs of all stages are concatenated and processed using a final 1×1 convolution to reduce the channel dimension to the desired number of classes.

For the instance segmentation branch, we implemented a Mask R-CNN [8]. We use a region proposal network to detect regions of interest, perform non-maximum suppression, execute ROI alignment, and predict 28×28 binary masks as well as class probabilities for each detected instance.

In order to combine the semantic and instance segmentation outputs, we use an internal differentiable fusion instead of external heuristics. For this purpose, we first select the most likely class label for each detected instance using a differentiable

$$\textit{soft argmax} = \sum_{i}^{N} \lfloor \frac{e^{z_i \cdot \beta}}{\sum_{k}^{N} e^{z_k \cdot \beta}} \rceil \cdot i \qquad (1)$$

operation [2], where N is the number of *things* classes, β is a large constant, and z is the predicted class logit. Using β in the exponent in combination with the round function allows us to squash all non-maxium values to zero. In this way, we approximate the non-differentiable *argmax* function, allowing us to backpropagate gradients.

We then resize the predicted 28×28 mask logits for each detected instance according to its predicted 2D bounding box size and place them in empty canvas layers at the predicted 2D location, as shown in Figure 2 (*top right*). Additionally, we merge the canvas layers for regions of interest with the same class id and high mask IOU. The resulting canvas collection from the instance segmentation branch is then concatenated with the *stuff* class logits of the semantic segmentation branch to generate our panoptic output, as illustrated in Figure 2 (bottom). The pixel-wise panoptic segmentation output is attained by applying a softmax layer on top of the stacked semantic and instance segmentation information. The shape of the final output is $H \times W \times (\#\,\textit{stuff classes} + \#\,\textit{detected instances})$. For *stuff* classes, the output is a class ID. For *things* classes, the output is an instance ID. The corresponding class ID for each instance can be gathered from our semantic or instance segmentation output.

During training, it is important to reorder the detected instances to match the order of the ground truth instances. For this purpose, we use a ground truth instance ID lookup table. All parameters of our network are optimized jointly.

3.2. Inter-task Relations

Our differentiable fusion of semantic and instance segmentation predictions allows us to join the outputs of our two branches internally for end-to-end training. However, it also allows us to provide instance predictions as additional feature input to our semantic segmentation branch, as shown in Figure 3.

For this purpose, we first evaluate our instance segmentation branch and build an instance canvas collection as described in Sec. 3.1. Next, we merge canvas layers of instances that belong to the same class using weighted average and insert empty canvas layers for missing or undetected classes. In this way, we generate an *initial segmentation image* (ISI) which represents a coarse semantic segmentation for *things* classes.

To exploit this segmentation prior in our semantic segmentation branch, we downsample our ISI to $H/4 \times W/4 \times$ *# things classes* and concatenate it with the output of our semantic segmentation upsampling modules, as shown in Figure 3. Next, we apply four network blocks consisting of 3×3 convolution, batch normalization, and ReLU followed by a single 1×1 convolution, batch normalization, and ReLU block to reduce the channel dimension to the number of classes. Finally, we use bilinear upsampling to obtain semantic segmentation logits at the original input image dimensions and apply a softmax non-linearity.

By exploiting the segmentation prior given by ISI, the upsampling modules of our semantic segmentation branch focus more on the prediction of *stuff* classes and boundaries between individual classes instead of *things* classes. This is a huge advantage compared to disjoint semantic and instance segmentation branches where redundant predictions are performed in the semantic segmentation branch. As a consequence, this link between the individual tasks increases the panoptic performance of our system.

4. Experimental Results

To demonstrate the benefits of our end-to-end panoptic architecture with interrelations, we evaluate it on the challenging Cityscapes dataset [4] for semantic understanding of urban street scenes. We follow the protocol of [4] and train and evaluate on 19 classes (11 *stuff* and 8 *things*). We use the recently introduced panoptic quality [11] metric to assess the segmentation performance.

4.1. Experimental Setup

Due to our limited computational resources, we limited the maximum number of instances per image to 30 and excluded samples with more instances from the evaluation. In this way, we use 2649 of 2975 training images ($\approx 89\%$) and 415 of 500 publicly available validation images ($\approx 83\%$). Additionally, we reduce the spatial image resolution from

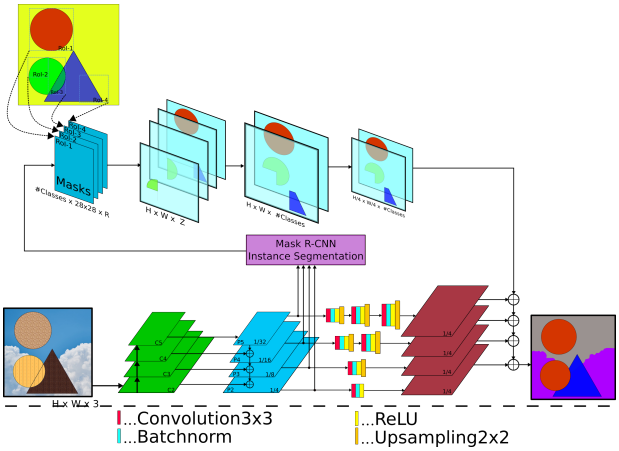


Figure 3: Illustration of our proposed semantic and instance segmentation branches with inter-task relations. We first run the instance segmentation branch and then provide instance segmentation predictions as additional feature input to the semantic segmentation branch via an initial segmentation image (ISI). Finally, we evaluate the semantic segmentation branch and exploit the segmentation prior given by ISI to improve the overall panoptic performance.

2048×1024 to 1024×512. For this reason, we cannot not benchmark against other state-of-the-art approaches. To provide an unbiased evaluation, we compare four different approaches with an increasing level of entanglement between semantic and instance segmentation. All methods use the same backbone, training protocol, and hyper-parameters:

Semantic + Instance. This approach uses two different networks based on a ResNet-101 [9] backbone which independently perform semantic and instance segmentation. A heuristic is used to combine the individual results.

Panoptic FPN. This method is a reimplementation of Panoptic Feature Pyramid Networks [11] with a ResNet-101 [9] backbone. In contrast to *Semantic + Instance*, the semantic and instance segmenation branches use a single shared feature representation. The results, however, are still merged heuristically.

HPS. Our holistic panoptic segmentation network (HPS) extends *Panoptic FPN* as described in Sec. 3.1. Our network internally builds the panoptic segmentation output using differentiable operations which enables us to optimize for the final objective.

HPS + ISI. This method augments our *HPS* with inter-task relations between the semantic and in-

Method	PQ	SQ	RQ	PQ^{Th}	SQ^{Th}	RQ^{Th}	PQ^{St}	SQ^{St}	RQ^{St}
Semantic + Instance	40.6	70.9	51.3	40.3	75.4	53.0	40.9	67.6	50.0
Panoptic FPN	41.9	73.7	53.4	43.0	75.2	56.6	41.2	72.5	51.1
HPS	42.9	74.5	54.3	43.4	75.7	56.7	42.6	73.6	52.5
HPS + ISI	**44.0**	**74.8**	**55.5**	**44.4**	**76.4**	**57.5**	**43.7**	**73.6**	**54.1**

Table 1: Quantitative results on the Cityscapes dataset. The results show that a shared feature backbone reduces overfitting compared to two disjoint networks (*Semantic + Instance* vs *Panoptic FPN*). Also, generating the final panoptic output internally and training the system end-to-end increases the performance (*Panoptic FPN* vs *HPS*). Finally, using inter-task relations in the form of an initial segmentation image (ISI) provides an effective segmentation prior and increases the overall panoptic quality as well as all other metrics (*HPS* vs *HPS + ISI*).

stance segmentation branches by using an initial segmentation image (ISI), as introduced in Sec. 3.2.

4.2. Results

The thus obtained results of the four methods described above on the Cityscapes dataset are summarized in Table 1. In addition, to the panoptic quality (PQ), we show the segmentation quality (SQ) and the recognition quality (RQ) for all classes, *things* (Th) classes only, and *stuff* (St) classes only. Since PQ is a measurement of semantic (SQ) and instance (RQ) segmentation quality an improvement in either part will increase the accuracy of the overall system.

Interestingly, *Semantic + Instance* performs worse than *Panoptic FPN*. We hypothesize that this is because the number of training images in Cityscapes is low. Thus, the shared feature backbone of *Panoptic FPN* acts as a regularizer which reduces overfitting compared to training two individual networks without shared features on this dataset.

Next, *HPS* improves upon *Panoptic FPN* across all metrics and classes, because we optimize for the final panoptic segmentation output. Our system minimizes a panoptic loss in addition to the semantic and instance segmentation losses which provides better guidance for the network. In this way, we do not rely on the heuristic merging of subtask predictions but directly generate the desired output internally which results in improved accuracy in practice.

Finally, *HPS + ISI* significantly outperforms all other methods because it additionally leverages inter-task relations. Compared to *Panoptic FPN*, *HPS + ISI* improves PQ by $+5\%$ relative from 41.9 to 44.0. Providing instance segmentation predictions as additional feature input for the semantic segmentation branch gives a segmentation prior. By exploiting this prior, the semantic segmentation branch can focus more on the prediction of *stuff* classes and boundaries between individual classes which results in improved accuracy across all metrics. Additionally, our architectural advances only add a neglible computational overhead during both training and inference compared to *Panoptic FPN*.

This quantitative improvement is also reflected qualitatively, as shown in Figure 4. We observe that *HPS + ISI* handles occlusions more accurately (1^{st} row) and resolves overlapping issues on its own while being less sensitive to speckle noise in semantically coherent regions (2^{nd} row). Thanks to our end-to-end training and inter-task relations, we predict more accurate semantic label transitions (3^{rd} row) and reduce confusion between classes with similar semantic meaning like *bus* and *car* (4^{th} row).

5. Conclusion

Panoptic segmentation is a challenging but important and practically highly relevant problem. As approaching panoptic segmentation by independently addressing semantic and instance segmentation has several limitations, we propose a single end-to-end trainable network architecture that directly optimizes for the final objective. Moreover, we present a way to share mutual information between the tasks by providing instance segmentation predictions as additional feature input for our semantic segmentation branch. This inter-task link allows us to exploit a segmentation prior and improves the overall panoptic quality. In this way, our work is a first step towards fully entangled panoptic segmentation.

Acknowledgment. This work was partially supported by the Christian Doppler Laboratory for Semantic 3D Computer Vision, funded in part by Qualcomm Inc.

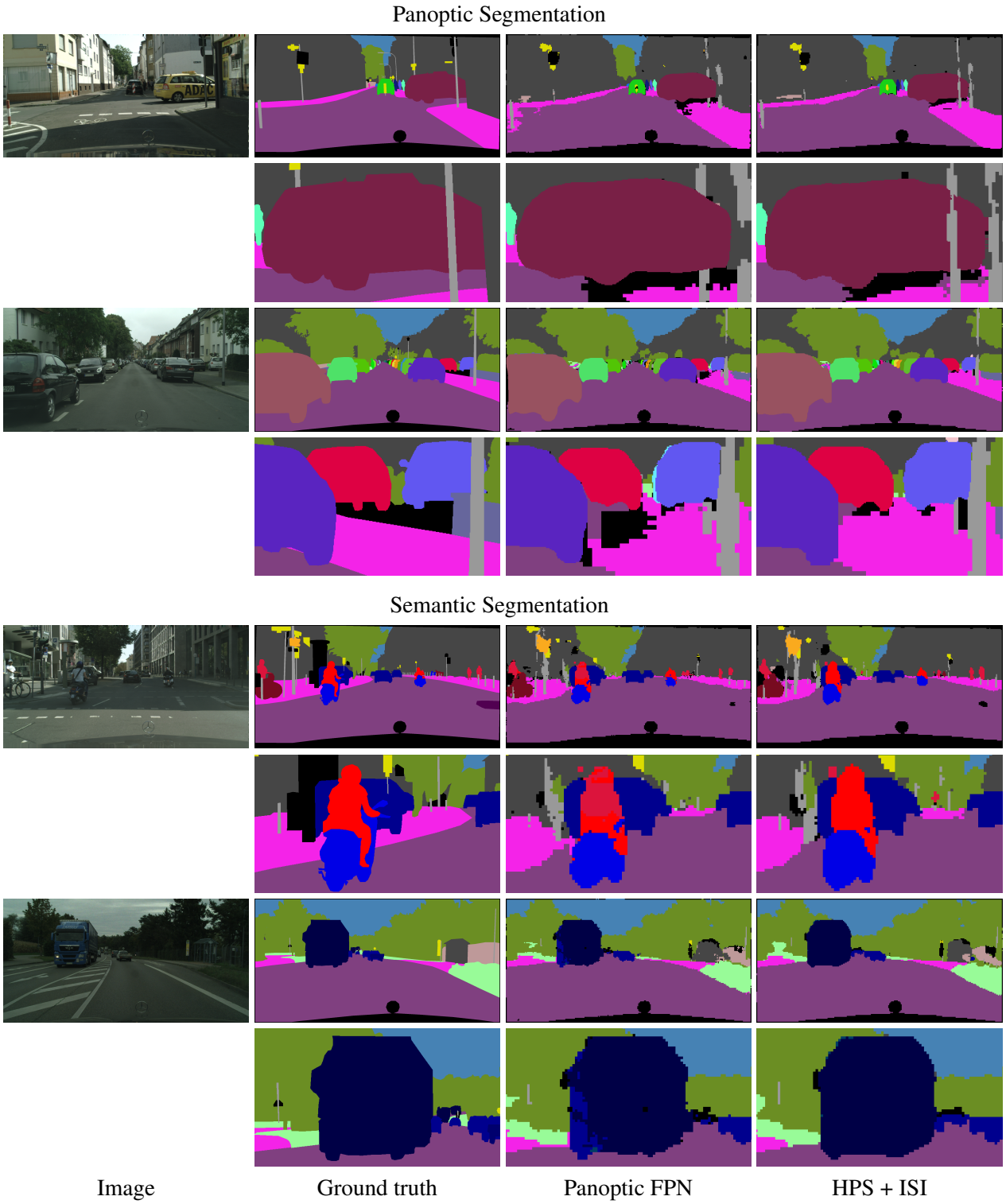


Figure 4: Qualitative results on the Cityscapes dataset. Compared to *Panoptic FPN*, *HPS + ISI* handles occlusions more accurately (1st row) and is less sensitive to speckle noise in semantically coherent regions (2nd row). Additionally, we predict more accurate semantic label transitions (3rd row) and reduce confusion between classes with similar semantic meaning like *rider* and *person* or *bus* and *car* (4th row). Both our end-to-end training as well as inter-task relations increase panoptic quality. **Best viewed in digital zoom.**

References

[1] COCO 2018 Panoptic Segmentation Task. http://cocodataset.org/index.htm#panoptic-leaderboard. Accessed: 2020-01-31.

[2] E. Brachmann, A. Krull, S. Nowozin, J. Shotton, F. Michel, S. Gumhold, and C. Rother. Dsac-differentiable ransac for camera localization. In *Conference on Computer Vision and Pattern Recognition*, pages 6684–6692, 2017.

[3] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam. Rethinking Atrous Convolution for Semantic Image Segmentation. *arXiv:1706.05587*, 2017.

[4] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The Cityscapes Dataset for Semantic Urban Scene Understanding. In *Conference on Computer Vision and Pattern Recognition*, pages 3213–3223, 2016.

[5] D. Feng, C. Haase-Schuetz, L. Rosenbaum, H. Hertlein, F. Duffhauss, C. Glaeser, W. Wiesbeck, and K. Dietmayer. Deep Multi-Modal Object Detection and Semantic Segmentation for Autonomous Driving: Datasets, Methods, and Challenges. *arXiv:1902.07830*, 2019.

[6] C.-Y. Fu, T. L. Berg, and A. C. Berg. IMP: Instance Mask Projection for High Accuracy Semantic Segmentation of Things. In *International Conference on Computer Vision*, pages 5178–5187, 2019.

[7] R. H. Hahnloser, R. Sarpeshkar, M. A. Mahowald, R. J. Douglas, and S. H. Seung. Digital Selection and Analogue Amplification Coexist in a Cortex-Inspired Silicon Circuit. *Nature*, 405(6789):947–951, 2000.

[8] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask R-CNN. In *International Conference on Computer Vision*, pages 2961–2969, 2017.

[9] K. He, X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In *Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[10] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv:1502.03167*, 2015.

[11] A. Kirillov, R. Girshick, K. He, and P. Dollár. Panoptic Feature Pyramid Networks. In *Conference on Computer Vision and Pattern Recognition*, pages 6399–6408, 2019.

[12] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollár. Panoptic Segmentation. In *Conference on Computer Vision and Pattern Recognition*, pages 9404–9413, 2019.

[13] A. Kirillov, E. Levinkov, B. Andres, B. Savchynskyy, and C. Rother. Instancecut: From Edges to Instances with Multicut. In *Conference on Computer Vision and Pattern Recognition*, pages 5008–5017, 2017.

[14] J. Li, A. Raventos, A. Bhargava, T. Tagawa, and A. Gaidon. Learning to Fuse Things and Stuff. *arXiv:1812.01192*, 2018.

[15] Y. Li, X. Chen, Z. Zhu, L. Xie, G. Huang, D. Du, and X. Wang. Attention-Guided Unified Network for Panoptic Segmentation. In *Conference on Computer Vision and Pattern Recognition*, pages 7026–7035, 2019.

[16] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei. Fully Convolutional Instance-Aware Semantic Segmentation. In *Conference on Computer Vision and Pattern Recognition*, pages 2359–2367, 2017.

[17] X. Liang, L. Lin, Y. Wei, X. Shen, J. Yang, and S. Yan. Proposal-Free Network for Instance-Level Object Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2978–2991, 2017.

[18] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature Pyramid Networks for Object Detection. In *Conference on Computer Vision and Pattern Recognition*, pages 2117–2125, 2017.

[19] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common Objects in Context. In *European Conference on Computer Vision*, pages 740–755, 2014.

[20] S. Liu, J. Jia, S. Fidler, and R. Urtasun. SGN: Sequential Grouping Networks for Instance Segmentation. In *International Conference on Computer Vision*, pages 3496–3504, 2017.

[21] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia. Path Aggregation Network for Instance Segmentation. In *Conference on Computer Vision and Pattern Recognition*, pages 8759–8768, 2018.

[22] J. Long, E. Shelhamer, and T. Darrell. Fully Convolutional Networks for Semantic Segmentation. In *Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[23] A. Petrovai and S. Nedevschi. Multi-Task Network for Panoptic Segmentation in Automated Driving. In *Intelligent Transportation Systems Conference*, pages 2394–2401, 2019.

[24] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention*, pages 234–241, 2015.

[25] J. Tighe, M. Niethammer, and S. Lazebnik. Scene Parsing with Object Instances and Occlusion Ordering. In *Conference on Computer Vision and Pattern Recognition*, pages 3748–3755, 2014.

[26] Z. Tu, X. Chen, A. L. Yuille, and S.-C. Zhu. Image Parsing: Unifying Segmentation, Detection, and Recognition. *International Journal of Computer Vision*, 63(2):113–140, 2005.

[27] J. Uhrig, M. Cordts, U. Franke, and T. Brox. Pixel-Level Encoding and Depth Layering for Instance-Level Semantic Labeling. In *German Conference on Pattern Recognition*, pages 14–25, 2016.

[28] Y. Xiong, R. Liao, H. Zhao, R. Hu, M. Bai, E. Yumer, and R. Urtasun. Upsnet: A Unified Panoptic Segmentation Network. In *Conference on Computer Vision and Pattern Recognition*, pages 8818–8826, 2019.

[29] Y. Yang, H. Li, X. Li, Q. Zhao, J. Wu, and Z. Lin. SOGNet: Scene Overlap Graph Network for Panoptic Segmentation. *arXiv:1911.07527*, 2019.

[30] D. Zhang, Y. Song, D. Liu, H. Jia, S. Liu, Y. Xia, H. Huang, and W. Cai. Panoptic Segmentation with an End-to-End Cell R-CNN for Pathology Image Analysis. In *Medical Image Computing and Computer-Assisted Intervention*, pages 237–244, 2018.

Frame-To-Frame Consistent Semantic Segmentation

Manuel Rebol Patrick Knöbelreiter
Graz University of Technology
rebol@student.tugraz.at, knoebelreiter@icg.tugraz.at

Abstract. *In this work, we aim for temporally consistent semantic segmentation throughout frames in a video. Many semantic segmentation algorithms process images individually which leads to an inconsistent scene interpretation due to illumination changes, occlusions and other variations over time. To achieve a temporally consistent prediction, we train a convolutional neural network (CNN) which propagates features through consecutive frames in a video using a convolutional long short term memory (ConvLSTM) cell. Besides the temporal feature propagation, we penalize inconsistencies in our loss function. We show in our experiments that the performance improves when utilizing video information compared to single frame prediction. The mean intersection over union (mIoU) metric on the Cityscapes validation set increases from 45.2% for the single frames to 57.9% for video data after implementing the ConvLSTM to propagate features trough time on the ESPNet. Most importantly, inconsistency decreases from 4.5% to 1.3% which is a reduction by 71.1%. Our results indicate that the added temporal information produces a frame-to-frame consistent and more accurate image understanding compared to single frame processing.*

1. Introduction

We address the task of semantic segmentation which assigns a semantic class for each pixel in an image. Our focus is on the computation of semantic segmentation for multiple consecutive images, referred to as frames, in a video sequence. Consecutive video frames contain similar information, because they capture a scene which only changes slightly. Therefore, the semantic segmentation of consecutive frames is similar as long as motion between frames does not increase significantly. For example, consider a street scene recorded by a camera mounted on a vehicle in which we observe a street sign. If the frame rate is large enough, we will observe the street sign in multiple images as the vehicle passes by. In this example, the goal of this work would be to consistently detect the street sign as such in all frames in which the sign appears. Single frame algorithms often fail at achieving this task. In general, we aim for temporally consistent segmentation of all semantic classes throughout a video sequence.

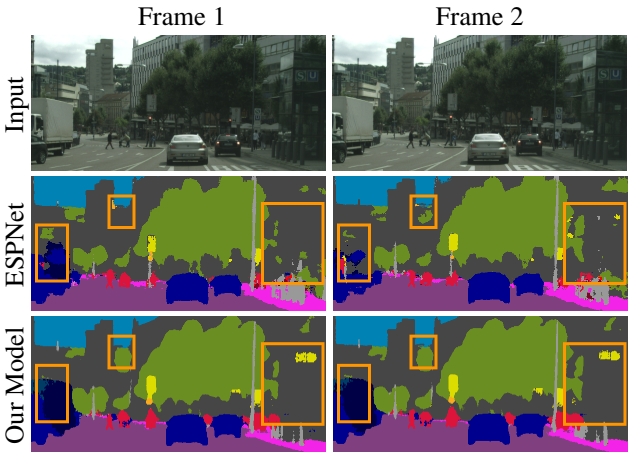


Figure 1: Consistent Semantic Segmentation. The trained ESPNet [19] model predicts temporally inconsistent semantic segmentation on two consecutive frames of the Cityscapes [6] video data set (second row). The semantic segmentation is color encoded and large inconsistencies are highlighted with orange boxes. The third row shows consistent results predicted by our model. We reduce temporal inconsistencies by 71%.

Many state of the art computer vision algorithms process images individually [26, 17, 3] and hence are not designed for video sequences. They do not consider the temporal dependencies which occur when segmenting a video semantically. If single frame convolutional neural networks (CNNs) predict semantic segmentation on video sequences, results can become temporally inconsistent because of illumina-

tion changes, occlusions and other variations. Figure 1 illustrates the differences between a temporally inconsistent prediction of video frames by a trained ESPNet [19] and our consistent model.

We address this issue by introducing methods which alter existing single frame CNN architectures such that their prediction accuracy benefits from having multiple frames of the same scene. Our method is designed such that it can be applied to any single frame CNN architecture. Potential applications include robotics and autonomous vehicles where video data can be recorded easily. Since we aim for a real-life application scenario our method does not access future frames. Instead, we only utilize information from past frames to predict the current frame. We implement our online method on the lightweight CNN architecture ESPNet. We include a recurrent neural network (RNN) layer into the ESPNet which allows past image features to be combined with current image features and thus computes consistent semantic segmentation over time. To train the parameters of our novel model for consistency, we introduce a inconsistency error term to our objective function. We verify our methods on a second architecture, which we name Semantic Segmentation Network (SSNet). The reason for the development of SSNet is to ensure that our methods do not only work on a specific CNN. We train the parameters of the two models on street scenes using supervised learning. The data is provided by the Cityscapes sequence [6] and a synthetic data set which we generate from the Carla [7] simulator. To avoid the large effort required to manually label video data, we use the pre-trained Xception model [3] to predict highly accurate video semantic segmentation.

2. Related Work

The best performances on semantic segmentation benchmark tasks such as PASCAL VOC [8] and Cityscapes [6] are reached by CNN architectures. Lightweight CNN architectures [19, 12, 29, 25, 27] have been developed to achieve high accuracy with low computational effort. We select the highly efficient ESPNet [19] as a basis for our work because it predicts semantic segmentation in real-time while maintaining high prediction accuracy. It uses point-wise convolutions together with a spatial pyramid of dilated convolutions [28]. The dilated convolutions allow the network to create a large receptive field while maintaining a shallow architecture. Although ESPNet processes images fast and accurately, it lacks temporal consistency when predicting consecutive frames. Therefore, we extend the ESPNet and enforce video consistency.

Video Consistency Kundu *et al*. [16] and Siddhartha *et al*. [2] base their work on the traditional graph cut [14, 15] approach towards semantic segmentation. They extent the traditional 2D to a 3D CRF by adding a temporal dimension which allows them to predict temporally consistent semantic segmentation on video. Compared to our approach additional optical flow information needs to be computed and the size of the temporal dimension must be predefined in advance. This results in additional computation complexity and less flexibility when changing parameters such as the frame rate. Therefore, we decided to implement RNNs [11, 23, 4] which offer a more flexible approach towards processing video data.

RNNs are trained to learn which features of past frames are relevant for current [18, 21] or future [24, 22] frames. In general, it is not clear if LSTM, GRU or any other RNN architecture is superior [5, 13, 10]. Depending on the application, one architecture might perform slightly better than the other [5]. Variations through modifying the proposed architectures might work even better in some cases [13]. The work of Jozefowicz *et al*. [13] shows the importance of the elements inside an RNN cell.

Lu *et al*. [18] use the plain LSTM to associate objects in a video. To enforce a frame-to-frame consistent prediction, they use an association loss during the training of the LSTM. Similarly, we implement a ConvLSTM and an inconsistency loss to tackle semantic segmentation. We place the ConvLSTM on different image feature levels in our architecture as suggested by [22, 21].

3. Consistent Video Semantic Segmentation

In this section, we introduce our methods towards frame-to-frame consistent semantic segmentation. We present different architecture to propagate features through time. To train the architectures for temporal consistency, we extend the cross entropy loss function with a novel inconsistency error term.

3.1. Temporal Feature Propagation

The propagation of image features from the past to the current time step allows the neural network to

make predictions based on time sequences. We prefer the ConvLSTM [23] cell for this dense prediction task. Compared to the fully connected LSTM, it removes unnecessary connections. For instance, the connection of features from the top left corner of the previous frame to features of the bottom right corner of the current frame is not needed. We assume that if we ensure consistency locally by the convolution operator, we will generate overall results which are consistent, as long as motion between frames can be detect in the local window. Therefore, we need to choose the filter size large enough to allow the ConvLSTM to detect local consistencies and motion between frames without explicit optical flow information. Furthermore, the ConvLSTM allows us to process images at different resolutions and reduces the number of parameters significantly compared to the fully connected LSTM. The definition of ConvLSTM cell is shown in [23]. We use two different networks in which we include the ConvLSTM cell. First, we introduce the Video SSNet (VSSNet) architecture which consists of six layers of 3×3 convolutions with dilation rates $\{1, 1, 2, 2, 4, 4\}$ and 64 channels. Compared to the SSNet, we replace the last convolutional layer with a ConvLSTM in the VSSNet. Second, we also extend the ESPNet [19] with a ConvLSTM layer. Although it would be reasonable to propagate features at every layer of a CNN architecture this is not feasible because of fast growing computational complexity. Figure 2 shows the ESPNet architecture with four possible positions for the ConvLSTM. The proposed architectures are enumerated alphabetically from ESPNet_L1a to ESPNet_L1d, starting with the ConvLSTM at the highest feature level which means that it is located closest to the output layer. Besides the ConvLSTM layer, we implement two ESP modules at the first spatial level and three ESP modules at the second spatial level, which is the simplest configuration introduced in [19]. All other aspects of the ESPNet architecture remain unchanged.

3.2. Temporal Consistency Loss

Our second building block to enforce consistency is an additional error term in our loss function. The resulting loss function $\mathcal{L}(\cdot)$ is defined as

$$\mathcal{L}(\mathbf{S}, \mathbf{P}) = \lambda_{\text{ce}}\mathcal{L}_{\text{ce}}(\mathbf{S}, \mathbf{P}) + \lambda_{\text{incons}}\mathcal{L}_{\text{incons}}(\mathbf{S}, \mathbf{P}), \quad (1)$$

where $\mathbf{S} \in \mathbb{S}^{T\times M\times N}$ contains the semantic ground truth and $\mathbf{P} \in \mathbb{R}^{T\times M\times N\times|\mathbb{S}|}$ contains the predictions. The set $\mathbb{S}$ contains all semantic labels. We bound the dimensions by the sequence length T, the image dimensions $M \times N$ and the number of semantic labels $|\mathbb{S}|$. The function $\mathcal{L}_{\text{ce}}(\cdot)$ computes the cross entropy loss and $\mathcal{L}_{\text{incons}}(\cdot)$ penalizes inconsistencies. The hyper-parameters λ_{ce} and λ_{incons} are introduced to influence the balance between training with focus on prediction accuracy or consistency.

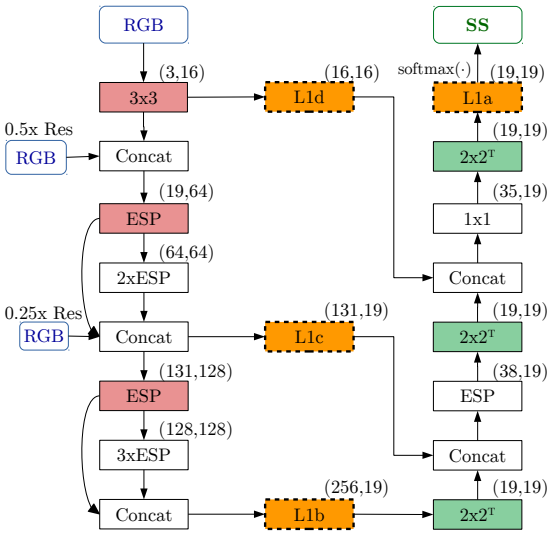


Figure 2: ESPNet with ConvLSTM. Four different positions for including a ConvLSTM (orange) into the existing ESPNet architecture are depicted. Dashed boxes indicate that only one ConvLSTM is present in a single architecture. L1b, L1c and L1d replace 1×1 channel reduction convolutions while L1a adds an additional layer to the architecture of the original ESPNet. Red boxes indicate a spatial dimensionality reduction by the factor two, while green boxes indicate a spatial dimensionality increase of two.

We define the inconsistency loss as

$$\mathcal{L}_{\text{incons}}(\mathbf{S}, \mathbf{P}) = \frac{1}{\omega_{\text{norm}}(\mathbf{S})} \sum_{t,m,n=1}^{T-1,M,N} \omega_{\text{vcc}}(\mathbf{S}, \mathbf{P}, t, m, n) \cdot \quad (2)$$
$$\left(\sum_{s=1}^{|\mathbb{S}|} \delta(\mathbf{S}_{t,m,n} = s) \cdot (\mathbf{P}_{t,m,n,s} - \mathbf{P}_{t+1,m,n,s})^2 \right),$$

where $\delta(\cdot)$ refers to the indicator function defined as

$$\delta(\phi(\cdot)) = \begin{cases} 1 & \text{if } \phi(\cdot) \text{ is true} \\ 0 & \text{else.} \end{cases} \quad (3)$$

The inconsistency loss penalizes pixels with different predictions in consecutive frames, which are already predicted correctly in at least one frame of the consecutive pair. This ensures that all other incorrect pixels are only affected by the cross-entropy loss. Additionally, $\delta(\mathbf{S}_{t,m,n} = s)$ selects only the correct semantic class for consistency enforcement. We nor-

malize by the sum of pixels which are valid and consistent in the ground truth. This is achieved by

$$\omega_{\text{norm}}(\mathbf{S}) = \sum_{t,m,n=1}^{T-1,M,N} \delta(\mathbf{S}_{t,m,n} \in \mathbb{S}) \cdot \delta(\mathbf{S}_{t,m,n} = \mathbf{S}_{t+1,m,n}), \tag{4}$$

where the first factor checks for validity and the second one consistency in the ground truth. The boolean function $\omega_{\text{vcc}}(\cdot)$ ensures that only *valid*, *consistent* and *correctly* (vcc) predicted pixels are affected by the following loss term.

$$\begin{aligned}\omega_{\text{vcc}}(\mathbf{S}, \mathbf{P}, t, m, n) = \delta(\mathbf{S}_{t,m,n} \in \mathbb{S}) \cdot \\ \delta(\mathbf{S}_{t,m,n} = \mathbf{S}_{t+1,m,n}) \cdot \\ \psi(\mathbf{S}_{t,m,n}, \mathbf{P}_{t,m,n}, \mathbf{S}_{t+1,m,n}, \mathbf{P}_{t+1,m,n}),\end{aligned} \tag{5}$$

where the first factor ensures validity, the second consistency and the third correct prediction in one of two consecutive images. The third factor is given by the boolean function $\psi(\cdot)$ which we define as

$$\begin{aligned}\psi(s_1, \mathbf{p}_1, s_2, \mathbf{p}_2) = \min(\delta(s_1 = \arg\max(\mathbf{p}_1)) + \\ \delta(s_2 = \arg\max(\mathbf{p}_2)), 1).\end{aligned} \tag{6}$$

This function determines for a pixel at a certain position if at least one prediction in the consecutive image pair is correct. The input parameters are given by the two prediction vectors $\mathbf{p}_1, \mathbf{p}_2 \in \mathbb{R}^{|\mathbb{S}|}$ and the two ground truth labels $s_1, s_2 \in \mathbb{S}$ for any pixel position. All four parameters are retrieved from $\mathbf{P}$ and $\mathbf{S}$.

In Figure 3 we point out pixels which are affected by the inconsistency loss. In the bottom right of the prediction, the road (purple) is labeled inconsistently. For these pixels the function $\omega_{\text{vcc}}(\cdot)$ returns true and they are penalized by the inconsistency loss.

4. Experiments

First, we explain the generation of semantic video data with ground truth and show the impact of synthetic data. Second, we evaluate our proposed methods, *i.e.* the feature propagation and the inconsistency loss.

Architectures and data preparation We use two models in our experiments, the ESPNet [19] and the SSNet. We train the models on images with half and quarter Cityscapes resolution to reduce computational complexity. Comparisons between different configurations are always trained for the same number of epochs which is chosen high enough to allow for convergence of the configurations. We generate the pseudo ground truth for the sequence validation set with the Deeplab Xception model [3].

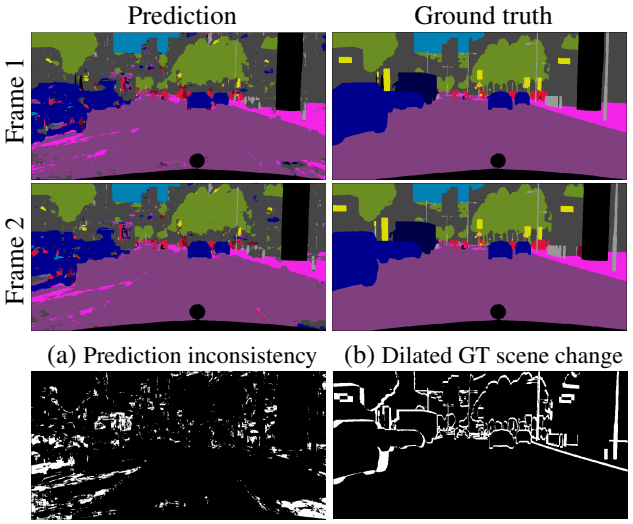


Figure 3: Visualization of Inconsistencies. We compare prediction and ground truth at two different time steps. The white pixels in image (a) are inconsistently predicted. Image (b) shows pixels which change their label because of motion. Only black pixels in image (b) are affected by our inconsistency loss.

Metrics and abbreviations The metrics which we use to compare our experiments are mean intersection over union (mIoU ↑), the percentage of correctly classified valid pixels (Acc ↑), the percentage of temporal consistently classified pixels (Cons ↑) and the percentage of pixels which are temporally consistent but wrongly classified (ConsW ↓). The arrow pointing upwards ↑ indicates that a higher value is better, whereas the arrow pointing downwards ↓ indicates the opposite. Our Cons and ConsW metrics check all pixels which need to have the same label according to the ground truth, *i.e.* black pixels in Figure 3b.

4.1. Data Generation

An important part of our work is the generation of ground truth for a video data set.We generate street scene video data with a pre-trained Deeplab Xception model [3] and the Carla simulator [7].

Real world data The semantic segmentation data sets of CamVid [1], Kitti [9], Cityscapes [6] and Mapillary [20] do not provide ground truth for video data because of the large labeling effort required. Therefore, we use the Deeplab Xception model pretrained on the Cityscapes data set to generate pseudo ground truth labels for the Cityscapes sequence data set. The reason why we prefer the Cityscapes dataset for video processing is that every 20th image of each

	Category	Experiment	mIoU	Acc	Cons	ConsW
	ESPNet	Single Frame	45.2	89.6	95.5	3.8
ConvLSTM	Convolution Type	ESPNet_L1a Std.	46.5	89.4	97.6	5.4
		ESPNet_L1a D.S.	45.2	89.0	97.2	5.5
	Position with Eq. Params.	ESPNet_L1a 7×7	50.3	91.4	98.5	3.1
		ESPNet_L1b 3×3	**52.0**	**91.5**	**98.7**	3.2
		ESPNet_L1c 5×5	49.9	91.4	98.2	3.0
		ESPNet_L1d 9×9	50.1	**91.5**	98.3	**2.9**

Table 1: ConvLSTM on ESPNet. Results on Cityscapes validation set. We compare the ESPNet trained with single frame images to different ConvLSTM configurations.

	Category	Experiment	mIoU	Acc	Cons	ConsW
Incons. Loss	Inconsistency Loss Func.	Sq Diff True	48.8	90.9	98.4	3.5
		Abs Diff True	48.6	90.9	98.6	3.5
	Inconsistency λ	$\lambda_{\text{incons}} = 0$	49.0	90.9	98.0	3.4
		$\lambda_{\text{incons}} = 10$	48.8	90.9	98.4	3.5
		$\lambda_{\text{incons}} = 100$	46.3	90.4	98.6	3.7
	Comb. Results ESPNet_L1b	**On Val. Set**	57.9	**93.0**	**98.7**	**2.7**
		On Test Set	**60.9**	-	-	-

Table 2: Top: Inconsistency Loss. We vary parameters of the loss function. Note that the inconsistency loss results cannot be compared directly to Table 1 because we only train the LSTM parameters for faster convergence. Bottom: Combined Results. The last two rows show the best results we are able to produce on Cityscapes validation and test set by combining the insights of our experiments.

sequence has annotated ground truth semantics. This allows for comparability with single frame results.

Synthetic data Besides Cityscapes data, we also generate synthetic data with the Carla simulator [7]. In total, we create 4680 scenes with 30 frames each. We train the ESPNet_L1b using different ratios between Cityscapes and Carla data. After training we evaluate on the Cityscapes sequence validation set. The quantitative results indicate that using about 10% synthetic data slightly improves frame-to-frame consistency (Cons) from 98.4% to 98.5% for Cityscapes only training while mIoU remains at 48.5%. When using more than 20% of synthetic data, mIoU on the Cityscapes validation set declines significantly. We assume the reason for the decline is that only 9 of 19 semantic classes are covered by Carla data. Nevertheless, we have shown that we can improve consistency by accurately labeled video semantic segmentation. For simplicity, we do not use the synthetic data set in other experiments.

4.2. Feature Propagation Evaluation

First, we compare different ConvLSTM as well as inconsistency loss configurations. Finally, we combine the insights from the comparison to achieve the highest performance.

ConvLSTM on VSSNet Training the VSSNet with ConvLSTM and inconsistency loss results in 44.6% mIoU, 89.9% Acc, 97.7% Cons and 4.7% ConsW. The results indicate that we are able to improve accuracy and consistency significantly, compared to the SSNet architecture trained with single frames which only achieves 39.9% mIoU and 94.4% Cons. After we have shown improvements on the VSSNet, we implement the following experiments on the ESPNet.

ConvLSTM configurations We test different convolution types and positions of the ConvLSTM as proposed in Figure 2. Table 1 shows the quantitative results of this comparison in the categories Convolution Types and Position with Equal Parameters. We compare the standard convolution operation with the depth-wise separable convolution inside the ConvLSTM on the ESPNet_L1a architecture. Results show that the standard convolution inside the ConvLSTM produces better results for all four metrics.

Furthermore, we evaluate the position of the ConvLSTM layer.We choose the filter size such that all experiments have a similar number of parameters for a fair comparison. This also ensures that the size of the receptive field at the layer is large enough to detect motion. The ESPNet_L1b architecture clearly outperforms all other architectures in both consistency and accuracy. This suggests that it is more efficient to propagate high level image features. Additionally, we found that the Parametric ReLU (prelu) performs better than the tanh activation function inside the ConvLSTM. Therefore, results are reported implementing the prelu activation function.

Inconsistency loss We test different inconsistency loss configurations on the ESPNet_L1b architecture because this model delivered the best results in previous experiments. Table 2 contains the quantitative results. We only train the LSTM parameters to allow for fast comparison of multiple models. The other parameters of the model are pretrained, but do not receive updates after the LSTM cell is added. Consequently, the scores are slightly lower than in Table 1. Substituting the squared difference loss inside Equation (2) with the absolute difference produces

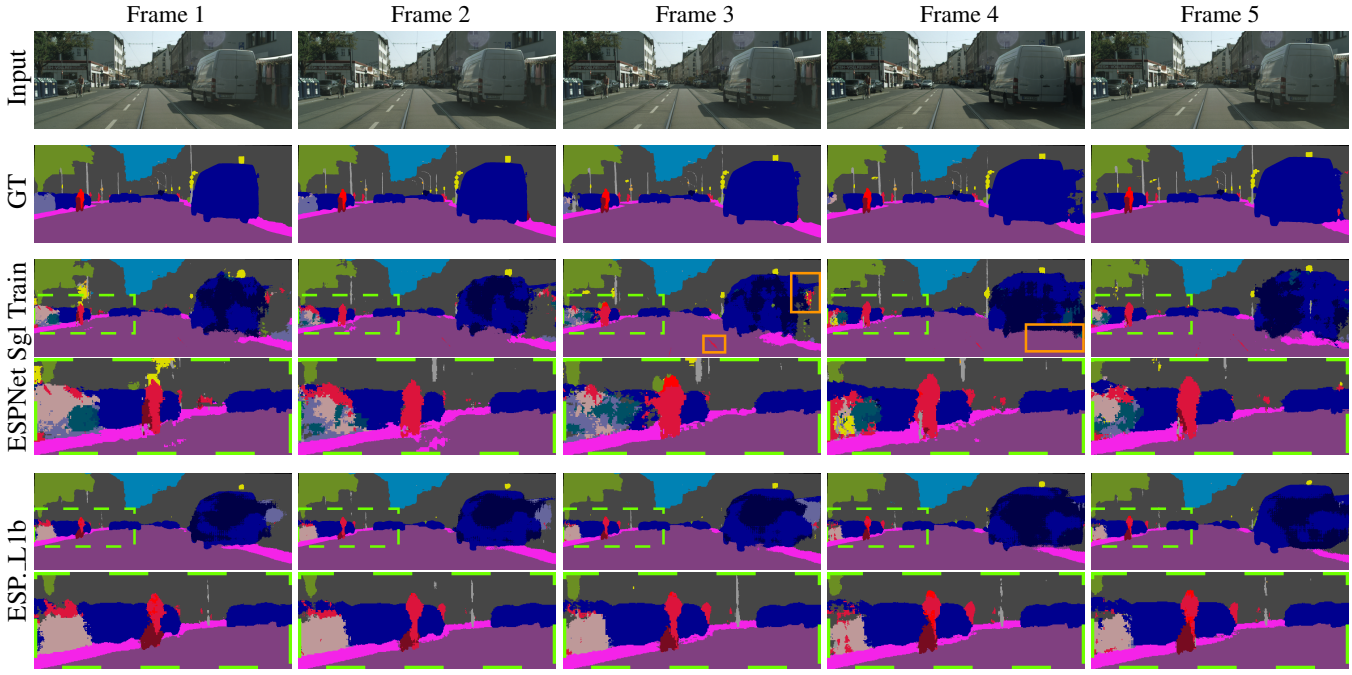


Figure 4: Qualitative Results. A comparison between input data, DeepLab Xception ground-truth, single frame training and LSTM training on the ESPNet (top to bottom). The horizontal axis represents the time steps. Areas with inconsistent predictions are shown in detail and highlighted with green dashed boxes. Other inconsistencies are highlighted with orange boxes. The ESPNet with single frame training (Sgl Train) produces inconsistencies in the right, left and on the road segmentation. The ESPNet_L1b predicts significantly more accurate and consistent results.

similar results. We observe that the hyper-parameter $\lambda_{\text{incons}} = 10$ provides a good trade-off between accuracy and consistency when using the squared difference loss function. The increase in consistency by 0.4 percentage points is noticeable when comparing the qualitative results. We set the other hyper-parameter $\lambda_{\text{ce}} = 1$ for all of our experiments.

Combining the findings In order to achieve the best results with ESPNet_L1b, we train the model in multiple phases. We use the squared difference inconsistency loss on correctly predicted classes with $\lambda_{\text{incons}} = 10$ and a 5×5 convolution inside the ConvLSTM. The quantitative results are shown at the bottom of Table 2. When training with the weighted cross entropy loss and data augmentations as proposed in [19] the official Cityscapes server reports 60.9% mIoU on the single frame test set. Our method reaches slightly higher accuracy and significantly better temporal consistency while using a similar number of parameters as Metha *et al*. [19].

5. Conclusion

We have shown that we can improve temporal consistency and accuracy of semantic segmentation for two different single frame architectures by adding feature propagation and a novel inconsistency loss. On the ESPNet, consistency and mIoU improve from 95.5 to 98.7% and from 45.2 to 57.9%, respectively. This is equal to a reduction of inconsistencies by 71.1% which can be observed immediately when watching a video sequence.

Moreover, we found that it is best to forward features at a high level with a standard convolution within the ConvLSTM cell. The hyper-parameter in our novel inconsistency loss function can be used to prioritize between consistency and accuracy. We also improve consistency slightly by adding synthetic data generated by the Carla simulator.

In future experiments we are interested in comparing other methods of adding the information from past frames to the current prediction. We also need to generate synthetic data such that it contains semantics of all validation classes to increase overall consistency and accuracy.

References

[1] G. J. Brostow, J. Fauqueur, and R. Cipolla. Semantic object classes in video: A high-definition ground

truth database. *Pattern Recognition Letters*, pages 88–97, 2009. 4

[2] S. Chandra, C. Couprie, and I. Kokkinos. Deep spatio-temporal random fields for efficient video segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8915–8924, 2018. 2

[3] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *European Conference on Computer Vision (ECCV)*, pages 833–851, 2018. 1, 2, 4

[4] K. Cho, B. van Merrienboer, Ç. Gülçehre, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *ArXiv*, 2014. 2

[5] J. Chung, C. Gülcehre, K. Cho, and Y. Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *ArXiv*, 2014. 2

[6] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3213–3223. 1, 2, 4

[7] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun. CARLA: An open urban driving simulator. In *Conference on Robot Learning (CoRL)*, pages 1–16, 2017. 2, 4, 5

[8] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, pages 98–136, 2015. 2

[9] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3354–3361, 2012. 4

[10] K. Greff, R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber. LSTM: A Search Space Odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, pages 2222–2232, 2017. 2

[11] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, pages 1735–1780, 1997. 2

[12] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *ArXiv*, 2017. 2

[13] R. Jozefowicz, W. Zaremba, and I. Sutskever. An empirical exploration of recurrent network architectures. In *International Conference on Machine Learning (ICML)*, pages 2342–2350, 2015. 2

[14] P. Krähenbühl and V. Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Proceedings of Advances in Neural Information Processing Systems*, pages 109–117, 2011. 2

[15] P. Krähenbühl and V. Koltun. Parameter learning and convergent inference for dense random fields. In *International Conference on Machine Learning (ICML)*, pages III–513–III–521, 2013. 2

[16] A. Kundu, V. Vineet, and V. Koltun. Feature space optimization for semantic video segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3168–3175, 2016. 2

[17] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *European Conference on Computer Vision (ECCV)*, pages 21–37, 2016. 1

[18] Y. Lu, C. Lu, and C.-K. Tang. Online video object detection using association lstm. In *IEEE International Conference on Computer Vision (ICCV)*, pages 2363–2371, 2017. 2

[19] S. Mehta, M. Rastegari, A. Caspi, L. G. Shapiro, and H. Hajishirzi. Espnet: Efficient spatial pyramid of dilated convolutions for semantic segmentation. In *European Conference on Computer Vision (ECCV)*, pages 561–580, 2018. 1, 2, 3, 4, 6

[20] G. Neuhold, T. Ollmann, S. Rota Bulò, and P. Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *IEEE International Conference on Computer Vision (ICCV)*, pages 5000–5009, 2017. 4

[21] C. Payer, D. Stern, M. Feiner, H. Bischof, and M. Urschler. Segmenting and tracking cell instances with cosine embeddings and recurrent hourglass networks. In *International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2018. 2

[22] S. shahabeddin Nabavi, M. Rochan, and Y. Wang. Future semantic segmentation with convolutional lstm. In *British Machine Vision Conference (BMVC)*, 2018. 2

[23] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Proceedings of Advances in Neural Information Processing Systems*, pages 802–810, 2015. 2, 3

[24] N. Srivastava, E. Mansimov, and R. R. Salakhutdinov. Unsupervised learning of video representations using lstms. In *International Conference on Machine Learning (ICML)*, pages 843–852, 2015. 2

[25] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI Conference on Artificial Intelligence*, pages 4278–4284, 2016. 2

[26] H. Touvron, A. Vedaldi, M. Douze, and H. Jégou. Fixing the train-test resolution discrepancy. *ArXiv*, 2019. 1

[27] S. Xie, R. B. Girshick, P. Dollár, Z. Tu, and K. He. Aggregated residual transformations for deep neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5987–5995, 2016. 2

[28] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. *ArXiv*, 2016. 2

[29] X. Zhang, X. Zhou, M. Lin, and J. Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6848–6856, 2018. 2

Ground Control Point Retrieval From SAR Satellite Imagery

Roland Perko, Hannes Raggam, Karlheinz Gutjahr
JOANNEUM RESEARCH Forschungsgesellschaft mbH, DIGITAL
{roland.perko,hannes.raggam,karlheinz.gutjahr}@joanneum.at

Wolfgang Koppe, Jürgen Janoth
Airbus Defence and Space
{wolfgang.koppe,juergen.janoth}@airbus.com

Abstract. *For many applications, like for instance autonomous driving or geo-referencing of optical satellite data, highly accurate reference coordinates are of importance. This work demonstrates that such Ground Control Points can automatically be derived from multi-beam Synthetic Aperture Radar satellite images with high accuracy.*

1. Introduction

Reliable Ground Control Points (GCPs), i.e., points of known geographical coordinates, are an essential input for the precise ortho-rectification of remote sensing imagery, the exact location of targets or the accurate geo-referencing of a variety of geo-datasets. Although GCPs collected by terrestrial means typically offer a high accuracy, their acquisition is expensive especially on a worldwide level.

Thus, a concept was formed to extract such GCPs from Synthetic Aperture Radar (SAR) satellite images (e.g., [9, 11]). Recently, refined SAR-based GCP extraction emerged due to three main reasons: (1) The 2D geo-location accuracy of current SAR sensors is very high, actually at centimeter level if atmospheric effects and Earth surface displacements are taken into account [5]. (2) Metallic objects like lamp poles or traffic signs (i.e., common features in urban scenes) appear as focused points in SAR images and can be detected with subpixel accuracy. (3) Using stereo acquisitions the 3D position (actually the ground mark) of these objects can be computed by means of radargrammetry.

Therefore, this work presents an automatic workflow, combining techniques from photogrammetric computer vision and remote sensing, that derives highly accurate GCPs from a set of multi-beam[1] high resolution images from TerraSAR-X, TanDEM-X, or PAZ satellites [3]. In contrast to [11], where persistent scatter interferometry (PSI) is deployed for point detection and 3D reconstruction, we build upon computer vision paradigms. Thus, the presented method can be efficiently applied on single images while PSI needs a stack of multiple images and is computationally very demanding [4]. In addition, our method can be applied on amplitude images alone as it does not rely on the phase information of the signal.

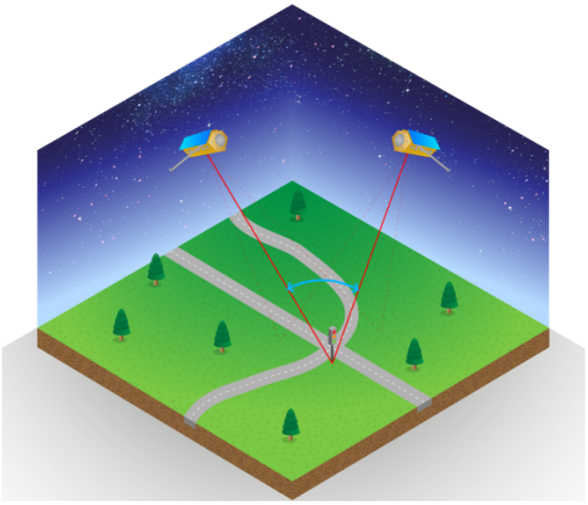


Figure 1. Stereo acquisition from space. Shown are two SAR satellites observing the same region on ground from two different orbital directions and look angles.

2. Method

The proposed fully automatic workflow for GCP retrieval consists of the following steps:

[1]The term *multi-beam* is equivalent to what is called *multi-view* in computer vision and stems from digital beamforming.

Image acquisition. Acquisition of a set of SAR images of the area of interest, in optimal case from ascending and descending orbital direction (cf. Figure 1). In case images are gathered from one orbital direction the stereo intersection angle has to be reasonably large (i.e., larger than 10°). After import each image consists of complex valued pixels plus the according sensor model (i.e., the cocircular geometry based on the Range and Doppler equations [2, 10]).

SAR delay correction. Adjustment of sensor models, in specific the SAR internal delays in range direction, for the following effects (cf. [5]): (1) Ionospheric signal propagation delay caused by electrons; (2) tropospheric signal propagation delay caused by air conditions, e.g., water vapor; (3) solid earth tides caused by gravity of moon and sun; and (4) plate tectonics, i.e., continental drift. For each image the range correction grid is updated, whereas the underlying information is gathered from weather and GPS services.

Point extraction. Metal objects appear as points or rather bright blobs on dark background (cf. Figure 2). For detection the image is upsampled based on complex FFT oversampling with a factor of 2. Then a matched filter is applied on the amplitude to localize blobs using a spike-shaped template kernel (cf. [14]). Results are thresholded and the best matching 2D blob locations are retrieved by subpixel interpolation [6, 12].

Matching of points. For each stereo pair epipolar rectified images using a coarse digital elevation model based on the method [13] are generated, also transferring the extracted points. This method undistorts the images in range direction and thus increases their geometric and radiometric similarity. Those points are then matched by means of normalized cross-correlation (kernel size depends on resolution of the input images). The resulting homologous points are then transformed back into the input images.

Retrieval of 3D coordinates. GCPs are calculated by a multi-image least squares spatial intersection of SAR range circles yielding a point cloud. Due to over determination incorrect points can be detected and rejected.

3. Results and Conclusion

The presented workflow was applied on a multitude of multi-beam scenes distributed over the whole globe, acquired with various imaging modes (i.e., Stripmap, Spotlight, HS Spotlight, Staring Spotlight [3]). Reference coordinates of metal poles were measured in-situ with differential GPS with an absolute 3D accuracy of ± 5 cm such that inaccuracies of the cadastre system do not propagate into the evaluation.

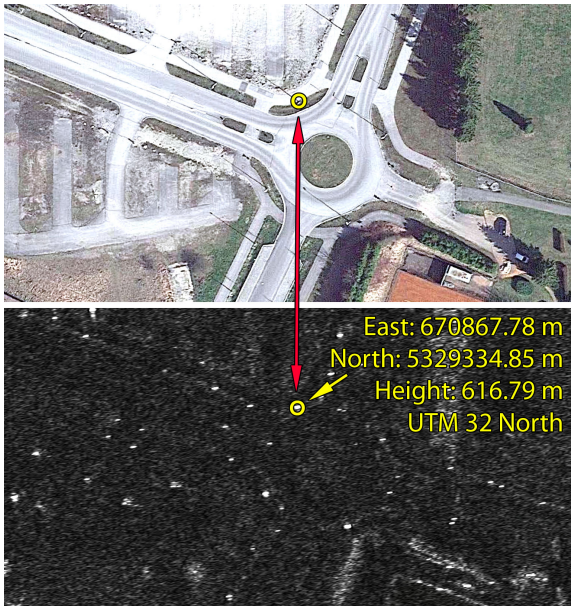


Figure 2. Roundabout traffic as perceived from an airborne digital camera (top) and from the SAR satellite (bottom). The bright blobs in the SAR amplitude corresponds mainly to light poles. An exemplary pole is highlighted together with its extracted 3D location.

Table 1 gives exemplary 3D accuracies (defined as root mean square (rms) values) as can be expected from the proposed methodology. In planimetry around 15 cm are achieved and in height around 20 cm, which are impressive numbers taking into account the altitude of the satellite's orbit at 514 km.

	East [m]	North [m]	Height [m]
rms	0.14	0.14	0.21
mean	0.04	0.09	0.07
std	0.13	0.10	0.20
min	-0.38	-0.26	-0.44
max	0.32	0.26	0.40

Table 1. 3D accuracy evaluation w.r.t. in-situ measurements given in meters based on 26 reference points and two opposite orbit Staring Spotlight images.

Future work will deal with automatic transfer of those SAR-based GCPs to optical images by means of multi-modal image matching. Most promising recent works use deep learning to tackle this ill-posed issue, for instance, [7, 8, 1].

References

[1] T. Bürgmann, W. Koppe, and M. Schmitt. Matching of TerraSAR-X derived ground control points to optical image patches using deep learning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 158:241–248, 2019.

[2] J. C. Curlander. Location of spaceborne SAR imagery. *IEEE Transactions on Geoscience and Remote Sensing*, (3):359–364, 1982.

[3] T. Fritz and M. Eineder. TerraSAR-X ground segment basic product specification document, TX-GS-DD-3302, issue 1.9. Technical report, DLR, 2013.

[4] S. Gernhardt, X. Cong, M. Eineder, S. Hinz, and R. Bamler. Geometrical fusion of multitrack PS point clouds. *IEEE Geoscience and Remote Sensing Letters*, 9(1):38–42, 2011.

[5] C. Gisinger, U. Balss, R. Pail, X. X. Zhu, S. Montazeri, S. Gernhardt, and M. Eineder. Precise three-dimensional stereo localization of corner reflectors and persistent scatterers with TerraSAR-X. *IEEE Transactions on Geoscience and Remote Sensing*, 53(4):1782–1802, 2014.

[6] S. S. Gleason, M. A. Hunt, and W. B. Jatko. Subpixel measurement of image features based on paraboloid surface fit. In *Machine Vision Systems Integration in Industry*, volume 1386, pages 135–144. International Society for Optics and Photonics, 1991.

[7] L. H. Hughes, M. Schmitt, L. Mou, Y. Wang, and X. X. Zhu. Identifying corresponding patches in SAR and optical images with a pseudo-siamese CNN. *IEEE Geoscience and Remote Sensing Letters*, 15(5):784–788, 2018.

[8] L. H. Hughes, M. Schmitt, and X. X. Zhu. Mining hard negative samples for SAR-optical image matching using generative adversarial networks. *Remote Sensing*, 10(10):1552, 2018.

[9] W. Koppe, R. Wenzel, S. Hennig, J. Janoth, P. Hummel, and H. Raggam. Quality assessment of TerraSAR-X derived ground control points. In *IEEE International Geoscience and Remote Sensing Symposium*, pages 3580–3583, 2012.

[10] F. W. Leberl. *Radargrammetric Image Processing*. Artech House, 1990.

[11] S. Montazeri, C. Gisinger, M. Eineder, and X. X. Zhu. Automatic detection and positioning of ground control points using TerraSAR-X multiaspect acquisitions. *IEEE Transactions on Geoscience and Remote Sensing*, 56(5):2613–2632, 2018.

[12] R. Perko. *Computer Vision for Large Format Digital Aerial Cameras*. PhD thesis, Graz, University of Technology, Austria, 2004.

[13] R. Perko, K. Gutjahr, M. Krüger, H. Raggam, and M. Schardt. DEM-based epipolar rectification for optimized radargrammetry. In *IEEE International Geoscience and Remote Sensing Symposium*, pages 969–972, 2017.

[14] R. Szeliski. *Computer Vision: Algorithms and Applications*. Springer Science & Business Media, 2010.

Classification and Segmentation of Scanned Library Catalogue Cards using Convolutional Neural Networks

Matthias Wödlinger, Robert Sablatnig
Computer Vision Lab, TU Wien
{mwoedlinger,sab}@cvl.tuwien.ac.at

Abstract. *The library of the TU Wien has been documenting changes in its inventory in the form of physical library archive cards. To make these archive cards digitally accessible, the cards and the text regions therein need to be categorized and the text must be made machine-readable. In this paper we present a pipeline consisting of classification, page segmentation and automated handwriting recognition that, given a scan of a library card, returns the category this card belongs to and an xml file containing the extracted and classified text.*

1. Introduction

A library catalogue is a register where all bibliographic entries found in a library are listed. In this paper we present a pipeline that automatically processes scanned images of library catalogue documents such that they can be made available and also searchable in an online database. While earlier work in this direction uses hand crafted rules and regular expressions to classify text in extracted OCR data, in recent years Convolutional Neural Network (CNN) based methods that operate on pixel level have formed the state-of-the-art in this task [4].

The library catalogue at hand consists of 113073 mostly handwritten documents, mostly collected in the time period from 1815 to 1930. The scanned images contain exactly the card with no surrounding content (see Fig. 1). Documents are classified into two groups: library cards with a "*Signatur*" (a unique identifier) that we call *S cards* and cards without it (*V cards*). V cards are not relevant for the online database and must be sorted out.

For training 2000 S cards and 500 V cards where manually extracted. The S cards where further sorted into 5 classes based on their layout. The text regions were manually annotated and verified by experts.

Model	Accuracy
ResNet18	0.988
ResNet34	0.988
ResNet50	**0.994**

Table 1. The accuracy scores on the test set. The accuracy is computed with respect to all 6 classes.

In this paper we describe a pipeline that, given a scanned library card image, determines if it is type S or V and then returns an xml file with the extracted and classified text. We describe the components of our pipeline in Section 2 and give a conclusion in Section 3.

2. Methodology and Results

The pipeline developed in this project is summarized in Fig. 1.

Classification of S and V cards We use a ResNet [2] pretrained on ImageNet and finetuned on our documents to sort out V cards. We do not freeze any layers during finetuning but instead train the full model with a smaller initial learning of $4 \cdot 10^{-4}$. To prevent large class imbalances we train the network on all 6 classes. The 2500 annotated documents are randomly split into train, test and validation sets and rescaled to 512×512. Table 1 shows the accuracy scores on the test set for three ResNets with different depth parameters.

Page segmentation of S cards The text regions in S cards are categorized in 7 classes that each contain document specific information like title, author, publisher or unique identifiers. The text region classes are distinguished from one another by location, font size and content. We use a CNN for image segmentation to detect and classify the text regions

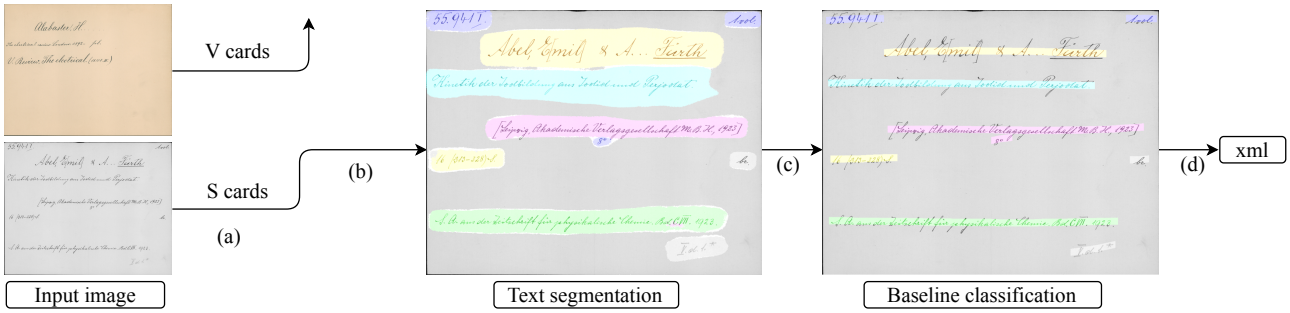


Figure 1. The proposed pipeline consisting of (a) an image classifier to sort out V cards (b) a segmentation network to detect and classify text regions and (c) baselines and finally (d) an HTR model whose output is combined with the baseline segmentation and saved as an xml file. Colors denote the different text categories.

Model	mIoU
Large Kernel Matters (ResNext101)	0.793
DeepLabV3+ (ResNet152)	**0.799**
dhSegment (ResNet50)	0.772

Table 2. The mIoU scores. The image classifiers in brackets denote the frontend used.

and later also the text baselines therein. We experiment with the models dhSegment [4], Global Convolutional Network (GCN) [5] and DeepLabV3+ [1]. The 2000 documents were first split in 50% train and 25% test and validation data each and then resized to 512×512. We found that adding a border around text regions (a line with constant width along the outline of text regions) as an additional class during training helps the network in learning to separate different text regions. Table 2 shows the mean intersection over union (mIoU) scores for the three best performing models. The segmentation is then used to classify the extracted text as described below.

Handwriting Recognition For the detection of text baselines and handwritten text recognition (HTR) model from Transkribus [3] are used. The Transkribus platform contains models for baseline detection and HTR pretrained on german Kurrent writing (with a character error rate of 7% on a seperate reference dataset [3]), which is the predominant writing style in our dataset. We apply the baseline detection of Transkribus, then classify the baselines according to the segmentation and add missing baselines for common errors. Afterwards the HTR model is applied and the result is saved as an xml file.

3. Conclusion

We have presented an approach for the automatic digitization of a library catalogue. We compared state-of-the-art models for semantic segmentation and found that DeepLabV3+ performs well in the task of page segmentation for historic handwritten documents. On the levels of baselines the classification of text using our segmentation appraoch performs reasonably well for the application however the character error rate of 7% needs improvement either through retraining on documents from our dataset or by manual corrections. For futher work, we believe that a better recognition of baselines has the largest potential for further improvements.

References

[1] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *European conference on computer vision (ECCV)*, pages 801–818, 2018.

[2] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[3] P. Kahle, S. Colutto, G. Hackl, and G. Mühlberger. Transkribus – a service platform for transcription, recognition and retrieval of historical documents. In *International Conference on Document Analysis and Recognition (ICDAR)*, volume 4, pages 19–24. IEEE, 2017.

[4] S. A. Oliveira, B. Seguin, and F. Kaplan. dhsegment: A generic deep-learning approach for document segmentation. In *International Conference on Frontiers in Handwriting Recognition (ICFHR)*, pages 7–12. IEEE, 2018.

[5] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun. Large kernel matters–improve semantic segmentation by global convolutional network. In *International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4353–4361, 2017.

Visual Odometry For Industrial Cable Laying

Ana Gregorac, Armin Köfler, Karlheinz Gutjahr, Richard Ladstädter, Roland Perko
JOANNEUM RESEARCH Forschungsgesellschaft mbH, DIGITAL, Austria
{firstname.lastname}@joanneum.at

Wolfgang Höppl
Ingenieurkonsulent für Vermessungswesen, Graz, Austria
w.hoeppl@geo-hoeppl.at

Abstract. *In order to support broadband network expansion in rural areas, the LAYJET Micro-Rohr Verlegegesellschaft has developed a highly automated cable laying technology based on a Fendt 936 tractor as the carrier vehicle and a milling machine with an integrated cable laying unit [3]. Operating at a speed of approximately 1kph, LAYJET is able to lay cables of several kilometres of length per day along of existing roads. The position of the cable needs to be precisely surveyed for documentation purposes, which is a time consuming and costly process. LAYJET is therefore equipped with a high-end GNSS RTK positioning system (TRIMBLE NetR9). In areas with bad GNSS signal reception or even complete GNSS outage (e.g., roads through a forest) an alternative positioning method is needed. JOANNEUM RESEARCH and the surveying office Höppl / Graz have therefore developed a calibrated stereo camera setup triggered by an odometer which allows reconstructing the trajectory of the GNSS antenna using visual odometry (VO).*

1. Introduction

VO is perfectly suited to reconstruct the trajectory of (very) slow moving vehicles as the LAYJET tractor as the drift error is dependent only on distance but not on time (as it is the case for Inertial Measurement Units (IMU)). Using a calibrated stereo camera system also allows determining the scale correctly without additional measurements [4]. In the following we describe the camera system and the implemented VO workflow and show first results from a LAYJET production run.



Figure 1. Example for a left and right camera view of the LAYJET camera system during operation.

2. Method

The stereo camera rig consists of a very stable steel bar carrying two camera housings separated by a baseline of 1.7m. The camera rig is mounted on top of the tractor at 3m height looking backward and tilted down by approximately 20 degrees. SONY Alpha 7 consumer 24 MPixel cameras equipped with 20mm lenses have been selected having a stable inner orientation in mind (auto focus can be switched off, no image stabilization). Calibration is done at the measurement lab of the Institute of Engineering Geodesy and Measurement Systems (IGMS) at TU Graz using the Remote Sensing Graz (RSG) software of JOANNEUM RESEARCH [2].

During field operation of the LAYJET system the stereo cameras are triggered at a fixed spacing of 2m $\pm$ 2cm by using the integrated odometer of the Fendt tractor (see Figure 1). This allows stable image trigger also in absence of a reliable GNSS solution. Images and GNSS positions (if available) are stored onboard and are transferred each day to a cloud storage from where they can be accessed in the surveying office for further processing.

Another software tool developed by JOANNEUM RESEARCH scans the data of each mission and decides for which sections the trajectory has to be im-



Figure 2. Template of the GNSS antenna (left) and best position found in the right image.

proved due to bad GNSS quality or if there are antenna positions missing due to GNSS outages and need to be derived from VO solely. It is important that there are GNSS positions available before and after an outage has occurred to ensure that the VO trajectory is correctly oriented and placed w.r.t. the defined coordinate system (UTM). One issue that had to be solved is to mask out all areas in the stereo images covered by the tractor or milling machine itself as this would deteriorate the VO process significantly and often caused complete fail of the VO solution.

The GNSS antenna is mounted straight above the position where the cable is laid at a known height offset. It is therefore necessary to determine the exact 3D position of the GNSS antenna which can move relatively to the camera system. This is solved by automated detection and measurement of the GNSS antenna in both stereo images using an advanced template matching process (see Figure 2). GNSS positions of good quality are introduced as ground control points (GCP) in the adjustment process. If the GNSS position is inaccurate or even unknown the 3D position of the antenna is reconstructed by using the stereo image measurements of that event.

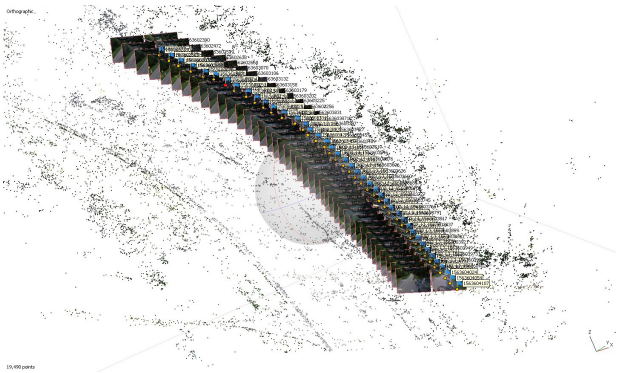
The VO workflow has been implemented by using the Agisoft Metashape v1.5.4 software [1] and its Python scripting capabilities. Importing of the images, correcting image distortions, applying image masks, feature point extraction, image matching and photogrammetric triangulation are fully automated by the script. The reconstructed camera positions and derived GNSS antenna positions can be inspected using the Metashape GUI and QC reporting tools. If the expected accuracy level has been reached the GNSS antenna position are exported to an ASCII coordinate file.

3. Results and Conclusion

The VO workflow has been tested with data from a LAYJET production run collected in Germany. The road passes through a forest which causes bad GNSS signal quality and a low number of visible satellites (in addition the RTK correction signal has been lost). The estimated position accuracy is therefore strongly reduced to about $\pm$ 2m. The photogrammetric bundle adjustment uses the GNSS solution as approximate positions and the well-defined relative geometry of the stereo pairs and consecutive stereo models to improve the accuracy at least by a factor of 10-20.

Figure 3 shows the reconstructed trajectory of the stereo rig and the derived GNSS antenna positions for 50 trigger events (section of 100m length). The sparse 3D point cloud generated during the VO process can be easily improved by an additional dense matching step which allows to inspect the environment and cable routing more closely.

First test evaluations have shown a throughput of about 10 stereo models per minute (50min per km) on an Intel workstation equipped with 16GB RAM and a NVIDIA GeForce GTX 1660 Ti GPU which should allow for overnight processing of the data collected on one day.


Figure 3. Path of the LAYJET tractor reconstructed using a VO workflow implemented in Metashape.

The VO system described in this paper derives absolute orientation angles solely from GNSS positions, which is straightforward for the heading angle but also works for roll and pitch as long as there are turns included in the trajectory. In case of exactly straight road sections the pitch angle is not defined and has to be set to zero. As the road cross profile inclination can be assumed to be in the range of $\pm$ 3deg this causes a lateral position error of up to 15cm (GNSS antenna height $\sim$ 2.5m).

For longer GNSS outages the estimation of the roll angle degrades with distance which can lead to significant height errors in case of steep descents. It is therefore recommended to integrate an additional inclinometer to measure roll and pitch angles at a precision of about $\pm$ 1deg in a next version of the LAYJET VO system.

References

[1] Agisoft LLC. *Agisoft Metashape User Manual, Professional edition, Version 1.5*, 2018 (accessed April 29, 2020). https://www.agisoft.com/.

[2] JOANNEUM RESEARCH. *Remote Sensing Software Graz*, 2018 (accessed April 29, 2020). https://www.remotesensing.at/remote-sensing-software.

[3] LAYJET Micro-Rohr Verlegegesellschaft. *Innovativer Glasfasernetzausbau: Wir verlegen Zukunft*, 2018 (accessed April 29, 2020). https://www.layjet.at.

[4] R. Mur-Artal and J. D. Tardós. ORB-SLAM2: An open-source SLAM system for monocular, stereo, and RGB-D cameras. *IEEE Transactions on Robotics*, 33(5):1255–1262, 2017.

Few-shot Object Detection Using Online Random Forests

Werner Bailer, Hannes Fassold
JOANNEUM RESEARCH, DIGITAL – Institute for Information and Communication Technologies
{werner.bailer,hannes.fassold}@joanneum.at

Abstract. *We propose an approach for few-shot object detection, consisting of a CNN-based generic object detector and feature extractor, and an online random forest as a classifier. This enables incremental training of the classifier, which reaches similar performance with around 20 samples as when using 50+ training samples in batch learning.*

1. Introduction

In many practical applications for object detection, it is relevant to detect new classes or subclasses of common objects, for which only very limited training data are available. While a large amount of literature on few-shot classification has been published in recent years, the problem of few-shot detection is more challenging, as it also involves identifying candidate regions for the yet unknown object classes. The problem of few-shot detection can be discriminated into the two following cases.

Refinement of existing classes. The new class to be trained is a specific subclass of a class already supported by an object detection algorithm, e.g., classifying "truck", when the classifier already has a class "vehicle". For this approach, an existing detector and classifier for the broader class (e.g. Yolo [6], Faster R-CNN [7]) can be used, and an additional classifier to be trained/adapted for the new classes is needed.

New classes. Candidate regions for such classes will not be found by the pretrained classifier, thus another detection approach is needed. One approach to find candidate regions is to use a detector trained on "objectness", i.e. the likelihood that a regions contains a coherent object. On the identified candidate regions feature extraction and classification can be performed, similar to the first case.

We aim to enable training new object classes with only few (i.e., 5-10) labeled examples, which may also not be available all at once, but being added gradually, improve the detector over time. The contribution of this paper is thus using a CNN-based object detection framework for generic object detection and feature extraction, and train an online classifier on these features. After discussing related work in Section 2, Section 3 presents the proposed approach and results, and Section 4 concludes the paper.

2. Related work

[1] does not actually perform detection, but uses bounding box regression as proposed in SSD to improve the localisation of the region of interest. Then binary object-or-not classification as proposed in Faster R-CNN is used, and uses a modified Faster R-CNN classifier to facilitate transfer learning. The work proposes regularisation based on the probability distribution of the known classes for the new target class. [2] propose a method for few-shot classification and detection, bootstrapped from few labeled instances. The method is based on components from Faster RCNN, using Selective Search or Edge Boxes for region proposals, and iteratively adds bounding box proposals and updates classifiers. [5] propose a pipeline using faster R-CNN up to ROI pooling, and two FC layers as feature extractors. Classification is then performed using a kernel method. [4] uses FPN to create an object detection pipeline using metric learning. Classification is done different for pretrained classes (using Inception v3 [10] up to FC2), while few-shot learning is done with FPN (in the DCN variant) instead. [9] propose to train a generic object detector on ImageNet, sampling positive and negative candidate regions. This approach is suitable for generic object detection, beyond the originally trained classes. An approach based on meta-features and learning reweighting of those fea-

tures is proposed in [3]. A recent work applies fine-tuning only region proposal and classification layers on a data set consisting of many base class and few new class samples while fixing the feature extraction part of the network can outperform meta-learning approaches [11].

3. Proposed Approach

We based our approach on [9], which we use as generic object detector and feature extractor. For the classification we follow the pipeline proposed in [12], which uses online random forests proposed in [8] as a classifier. The random forest can be incrementally trained, and is able to provide good results with few training samples. We use the model pre-trained on ImageNet from [9], and evaluate it on the 12 classes dataset provided with the authors' implementation[1]. Each of the classes has between 55 and 108 training samples. We compare to a linear classifier trained on the entire set of samples, and train our online random forest based classifier with all or a fixed subset of samples per class.

With the full set of examples, the online random forest based classifier performs similarly but slightly worse than the linear classifier, with an F1 score of about 0.80. Down to about 20 samples per class, the performance stays nearly constant. With 10 samples the performance drops to around 0.70, with 5 samples to about 0.67. Only then the performance starts to degrade more quickly, arriving at only about 1.5 times better than random when using a single sample. The results are visualised in Figure 1. It is apparent, that the reduction of the F1-score is mainly due to reduced recall. In nearly all cases the loss in terms of recall is caused by misclassifying the object, while only in few cases the target object is missed in the detection stage.

4. Conclusion

Based on a recently proposed framework, which we use for generic object detection and feature extraction, we have developed an approach for few-shot object detection using an online random forest as a classifier, which makes it incrementally trainable. With about 20 samples there performance in terms of F1 score is similar to a linear classifier on the full set, and drops by about 0.13 when using only 5 samples, which makes this a practically usable approach in use cases with few training samples.

[1] https://github.com/mahyarnajibi/SNIPER/tree/cvpr3k

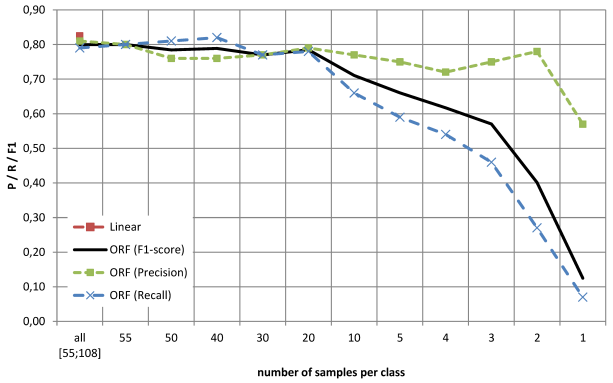


Figure 1. Detection results (F1 score, precision, recall) of the proposed approach on the 12 classes data set from [9], when trained on different numbers of samples per class. The confidence threshold is 0.15 for the online random forest classifier.

Acknowledgments

This work has received funding from the European Union's Horizon 2020 research and innovation programme, under grant agreements n° 761802 MARCONI ("Multimedia and Augmented Radio Creation: Online, iNteractive, Individual") and n° 761934, Hyper360 ("Enriching 360 media with 3D storytelling and personalisation elements").

References

[1] H. Chen, Y. Wang, G. Wang, and Y. Qiao. Lstd: A low-shot transfer detector for object detection. In *32nd AAAI Conf. on AI*, 2018.

[2] X. Dong, L. Zheng, F. Ma, Y. Yang, and D. Meng. Few-example object detection with model communication. *IEEE T.PAMI*, 41(7), 2018.

[3] B. Kang, Z. Liu, X. Wang, F. Yu, J. Feng, and T. Darrell. Few-shot object detection via feature reweighting. In *Proc. ICCV*, 2019.

[4] L. Karlinsky, J. Shtok, S. Harary, E. Schwartz, A. Aides, R. Feris, R. Giryes, and A. M. Bronstein. Repmet: Representative-based metric learning for classification and few-shot object detection. In *Proc. CVPR*, 2019.

[5] E. Maiettini, G. Pasquale, L. Rosasco, and L. Natale. Speeding-up object detection training for robotics with falkon. In *RSJ IROS*, 2018.

[6] J. Redmon and A. Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[7] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.

[8] A. Saffari, C. Leistner, J. Santner, M. Godec, and H. Bischof. On-line random forests. In *Proc. ICCV Workshops*, 2009.

[9] B. Singh, H. Li, A. Sharma, and L. S. Davis. R-FCN-3000 at 30fps: Decoupling detection and classification. In *Proc. CVPR*, 2018.

[10] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *Proc. CVPR*, 2016.

[11] X. Wang, T. E. Huang, T. Darrell, J. E. Gonzalez, and F. Yu. Frustratingly simple few-shot object detection. *arXiv preprint arXiv:2003.06957*, 2020.

[12] M. Winter and W. Bailer. Incremental training for face recognition. In *Intl. Conf. on Multimedia Modeling*. Springer, 2019.

The Problem of Fragmented Occlusion in Object Detection

Julian Pegoraro[1], Roman Pflugfelder[1,2]
[1] AIT Austrian Institute of Technology, [1,2] TU Wien
{julian.pegoraro|roman.pflugfelder}@ait.ac.at, roman.pflugfelder@tuwien.ac.at

Abstract. *Object detection in natural environments is still a very challenging task, even though deep learning has brought a tremendous improvement in performance over the last years. A fundamental problem of object detection based on deep learning is that neither the training data nor the suggested models are intended for the challenge of fragmented occlusion. Fragmented occlusion is much more challenging than ordinary partial occlusion and occurs frequently in natural environments such as forests. A motivating example of fragmented occlusion is object detection through foliage which is an essential requirement in green border surveillance. This paper presents an analysis of state-of-the-art detectors with imagery of green borders and proposes to train Mask R-CNN on new training data which captures explicitly the problem of fragmented occlusion. The results show clear improvements of Mask R-CNN with this new training strategy (also against other detectors) for data showing slight fragmented occlusion.*

1. Introduction

Automated surveillance at green borders has become a hot topic for European border guards. Border guards today face several challenges in protecting EU borders. One well known occasion in public is illegal migration which had its peak in 2015.

Border surveillance today limited to 2D imaging sensors consists of color and thermal cameras, mounted on poles or used as handheld cameras by the border guards. Innovating these technical systems by adding further capabilities of automatic inference such as the automatic detection of persons, vehicles, animals and suspicious objects in general will need to apply object detectors to such imagery.

However, video of green borders especially at EU borders show significant differences to typical imagery of video surveillance such as indoor video or video taken in man-made outdoor scenes. For example, green borders are scenes showing dense forest, hills, harsh weather and climate conditions. Such scenes draw challenges to automated surveillance and raise several interesting research questions.



Figure 1: The problem of fragmented occlusion in object detection. Top Left: no occlusion (level L_0). Top Right: slight occ. (L_1). Bottom Left: moderate occ. (L_2). Bottom Right: heavy occ. (L_3) occlusion.

This paper considers a challenge for state-of-the-art object detection in green border surveillance which is the problem of through foliage detection. To the best of our knowledge, none of the current approaches for object detection allow the detection of objects through foliage. This problem raises an interesting scientific question, namely how to detect objects with fragmented occlusion? This problem is also different to the problem of partial occlusion in

object detection. Fragmented occlusion occurs by viewing objects behind tree ans bush leaves. Contrary to partial occlusion, fragmented occlusion gives no clear view on minimal recognisable parts of the object [10] which is used to detect the object [7].

We show in this work that the state-of-the-art in object detection fails on fragmented occlusion even for the moderate case. For this, we created a new dataset (Figure 1) capturing people behind trees. We labelled nearly 40,000 images in three representative videos. This data raises new challenges on the labelling and evaluation which we only partially answer in this paper. For example, bounding boxes are the standard in current evaluation of detectors but such labels are hard to find in data that contains fragmented occlusion. As the state-of-the-art detectors deliver bounding boxes, fragmented occlusion poses new questions on the evaluation methodology.

Furthermore, we augmented Microsoft COCO[1] training data by occluding the ground truth masks similarly as leaves occlude people behind bushes and trees. We then show results on training Mask R-CNN [4] on this new data showing improvement of Mask R-CNN trained on the original data with slight fragmented occlusion.

2. Related Work

State-of-the-art object detection is based on deep learning. Two-stage detectors work by finding as an intermediate step bounding box proposals [3, 2] on the feature maps of the backbone CNN. A region proposal network further improves efficiency [9, 4]. One-stage detectors regress the bounding boxes directly [8, 6] which is computationally efficient on GPUs but this approach is inherently less accurate as it assumes a coarsely discretised search space. Although these methods show usually excellent performance for fully visible objects, they break down in the case of fragmented occlusion. Fragmented occlusion has not been considered for object detection so far, however there is literature about this topic in the field of motion analysis [1].

3. Methodology

We created a dataset recorded in a forest consisting of three videos with a total of 18,360 frames and 33,933 bounding boxes which were manually defined by human annotators. These bounding boxes are divided into four different occlusion levels including the unoccluded case (Figure 1).

[1] `http://cocodataset.org`



Figure 2: A training image from Microsoft COCO (`http://images.cocodataset.org/train2017/000000001700.jpg`). Top Left: the image. Top Right: Segmentation mask of the image. Bottom Left: image overlaid with artificial trees. Bottom Right: Mask of the overlaid image.

Then, we extended the Microsoft COCO dataset by adding artificial trees as foreground to the images of objects (Figure 2). We chose this dataset, because it contains pixel-wise segmentation masks in the ground truth as well as a large number of different categories including the human person.

The underlying basic idea of our approach is to add artificial fragmented occlusion to Microsoft COCO and train Mask R-CNN on this new data. By this we can adapt the original distribution of data to the case of fragmentally occluded objects. Since we are only interested in humans, we apply this augmentation only to images containing humans and use only these images for training. The trees used for the augmentation are generated from real images we have obtained from the test data. The method generates whole artificial trees by randomly adding branches to previously manually segmented tree trunks. In total 14 such trunks are extracted from the test dataset. The branches attached to these trunks are also randomly generated by also adding a few manually segmented leaves.

The trees are placed in front of objects by randomly selecting the x-coordinate on which they will be placed and an angle at which the tree will be rotated. The calculated foreground is applied to the image and its negative mask is multiplied by the segmentation mask of the objects in the image. The Mask R-CNN model is then trained with the aug-

mented images. The selected backbone model is the Inception v2 [5] network. This network is selected for its faster computation.

4. Evaluation

To evaluate whether training with the augmented dataset is useful, the model trained on the augmented data must be compared with the model not trained on this data. However, the intersection over Union (IoU) measure is not meaningful in this case.

Standard evaluation metrics such as the mean average precision (mAP) define an IoU threshold (e.g. 0.5) and check whether a ground truth object and a detected object have an IoU value above this value. If this is the case, the detected object is defined as a True Positive (TP). If an object is detected but there is no respective ground truth with an IoU above this specific threshold, the detected object is defined as a False Positive (FP). If there is ground truth but no detected object with an IoU above the threshold, the object is defined as a False Negative (FN).

These evaluation methods cannot be easily applied to ground truth showing fragmented occlusion, because of the following two observations:

IoU too small: Since the data is based on fragmented detections, a detector can only detect parts of the person. An image where this problem occurs is shown in Figure 3. The bounding box is clearly a TP, based on the fact, that fragmented objects should be detected, but due to the occlusion by the branches of the tree, the whole body cannot be recognized. This leads to an IoU of only ≈ 0.2.

Multiple detections: Another major problem with the standard evaluation metrics is that exactly one detected bounding box and one ground truth bounding box match. However, when handling fragmented objects, human heads and/or other body parts should be detected separately if body parts are covered. This creates the problem that parts of the body (like a head) is detected as well as the whole body. Figure 4 shows some examples.

To tackle these two problems, this paper proposes a different evaluation metric. For each bounding box in the evaluation data set, we calculate the maximum region in the image where there is no overlap with another ground truth bounding box. This region is then extracted and fed into the model. If the model detects an object, we define it as TP, otherwise as FN. To assess FPs, we create an additional dataset that represents the maximum region in an image without



Figure 3: Ground truth (green) and the detection (blue) vary substantially due to the occlusion effects.

overlap with any ground truth bounding box. We extracted in total 45,340 such regions with different aspect ratios, different parts of the image and at different time instants. In addition to FPs, we can also calculate the TNs using this evaluation metrics.

Figure 5 shows these results as recall vs. precision curve (ROC). There is no significant difference between Mask R-CNN trained on Microsoft COCO and on the augmented dataset for L_0 occlusion. However, clear improvement has been achieved for L_1 and L_2 occlusion which proves the applicability of the idea to model fragmented occlusion by the masks. Nevertheless, all approaches basically do not reach the expected robustness and accuracy for moderate L_2 and heavy L_3 occlusion. One reason for this is that our current technique is not accurate enough to model fragmented occlusion. Furthermore, clear limits exist as heavy fragmented occlusion removes local spatial and structural information necessary for current approaches in object detection.

We further recognise that bounding box labelling is not the appropriate approach for labelling data showing fragmented occlusion. Especially for L_3 and L_4 occlusion, it is frequently impossible to manually define the bounding box. Such occlusion levels allow an approximate localisation of the object in the image but make the observation of the object's extent impossible. While the recall in Figure 5 is still meaningful, the precision is basically undefined. This observation has severe consequences on the labelling,

Figure 4: The problem of multiple detections. Ground truth is shown in green. Left: state-of-the-art yields two bounding boxes of the same, single person. Middle: two persons are visible. Detection yields two bounding boxes which are diffucult to associate. Right: an even harder case with three persons.



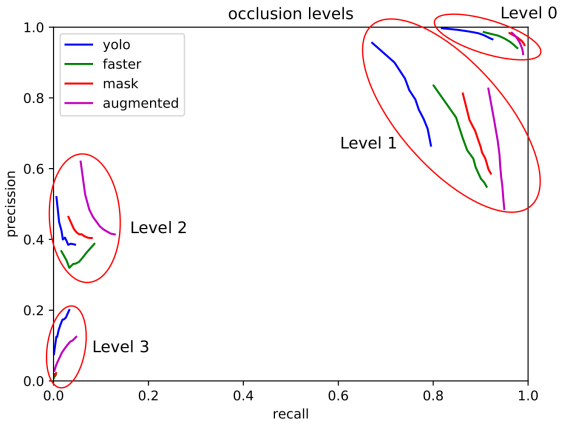


Figure 5: This ROC plot shows results of Faster R-CNN (green), YOLO (blue), Mask R-CNN (red) and our method (purple) for all occlusion levels.

but also on the evaluation and on the detector which we leave open for future research.

5. Conclusion

This paper formulates a new scientific question on object detection with fragmented occlusion which is different to partial occlusion. We show by a study that current object detectors fail in this case. We generated and labelled a new dataset showing people behind trees in a forestry environment. Such scenes frequently occur in border surveillance which has become very important in EU security policies. We try to tackle the occlusion challenge by augmenting Microsoft COCO including the pixel-wise segmentation masks to capture the occlusion problem. We show that Mask R-CNN trained on this data improves on fragmented occlusion, however, we also observe severe loss of spatial, structural information and that the bounding box itself is not the appropriate description to cope with fragmented occlusion. This has severe implications on the detection approach itself, but also on dataset labelling and evaluation. A potential solution is left open for future work.

Acknowledgments

This research was supported by the European Union H2020 programme under grant agreement FOLDOUT-787021. We thank all our students on internship to label the new dataset.

References

[1] M. Black and P. Anandan. The robust estimation of multiple motions: Parametric and piecewise-smooth flow fields. *Computer Vision and Image Understanding*, 63:75–104, 01 1996.

[2] R. Girshick. Fast r-cnn. In *CVPR*, pages 1440–1448, 2015.

[3] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, pages 580–587, 2014.

[4] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *CVPR*, pages 2961–2969, 2017.

[5] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[6] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *ECCV*, pages 21–37. Springer, 2016.

[7] G. Nebehay and R. Pflugfelder. Clustering of static-adaptive correspondences for deformable object tracking. In *CVPR*, June 2015.

[8] J. Redmon and A. Farhadi. Yolov3: An incremental improvement. *arXiv*, 2018.

[9] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, pages 91–99, 2015.

[10] S. Ullman, L. Assif, E. Fetaya, and D. Harari. Atoms of recognition in human and computer vision. *PNAS*, 113(10):2744–2749, 2016.

A Centerline-Guided Approach for Aorta and Stent-Graft Segmentation

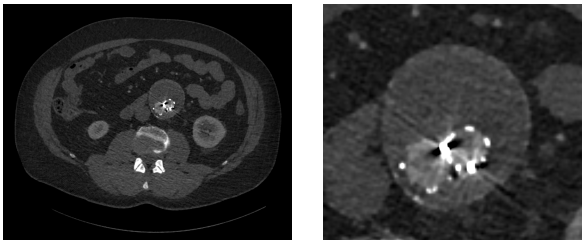
Bertram Sabrowsky-Hirsch, Stefan Thumfart, Richard Hofer, Wolfgang Fenz,
Research Unit Medical Informatics, RISC Software GmbH, Johannes Kepler University, Austria
{bertram.sabrowsky-hirsch,stefan.thumfart,richard.hofer,
wolfgang.fenz}@risc-software.at

Pierre Schmit, Franz Fellner
Central Radiology Institute, Kepler University Hospital, Austria
{pierre.schmit,franz.fellner}@kepleruniklinikum.at

Abstract. *Monitoring of patients after Endovascular aortic repair (EVAR) is a clinical necessity due to the high re-intervention rate associated with the treatment. The risk assessment could be greatly enhanced by the inclusion of metrics based on the aortic blood-flow and stent-graft changes. A preliminary step to this endeavour is, however, the automatic reconstruction of the relevant structures: aortic blood-lumen and the stent-graft wire frame. In this paper we present a centerline-guided approach that leverages knowledge about the target structures through a combination of two 3D U-Nets for efficient automated segmentation of both structures. We evaluate our approach on a real-world clinical dataset yielding Dice similarity coefficients of 0.942 and 0.841 for the blood lumen and stent-graft metal wire, respectively.*

1. Introduction

The abdominal aorta is the largest artery in the human body, with the descending branch supplying the lower body with about 4 liters of blood per minute [2]. Abdominal Aortic Aneurysms (AAAs) are critical as a rupture causes massive blood loss that quickly leads to death at a mortality rate of 85% to 90% [11], with half of the patients succumbing before they reach a hospital [1]. Overall, AAAs account for $175\,000$ deaths per year globally [7]. In contrast to open surgery, endovascular aortic repair (EVAR) poses a minimally invasive alternative that significantly reduces the intraoperative stress on the patients, who in turn experience shorter periods of convalescence. As a result, EVAR is the treatment of choice for 60% of patients [3]. These advantages come, however, at the cost of a high re-intervention rate of 20% [18], necessitating post-operative monitoring of the patients. We seek to aid monitoring by automatically calculating risk factors from blood flow simulations, which require prior segmentation of the target structures. In this paper, we present a novel method for segmenting the aortic blood lumen and the stent graft wire frame from post-operative abdominal CT-A scans.



(a) (b)

Figure 1. An abdominal CT-A scan (a) and a close-up view of imaging artifacts caused by the stent-graft wire frame (b). The wire frame of the Medtronic Endurant stent-graft encompasses the blood lumen in the two iliac bifurcations and is itself surrounded by the thrombosis.

2. Related Work

Blood vessel segmentation is an active field in research [19] and the variety of approaches reflects the diversity of both the targeted anatomical regions and the available imaging modalities. For clinical monitoring of the abdominal aorta after EVAR, CT-A is the modality of choice [22]. However, unique

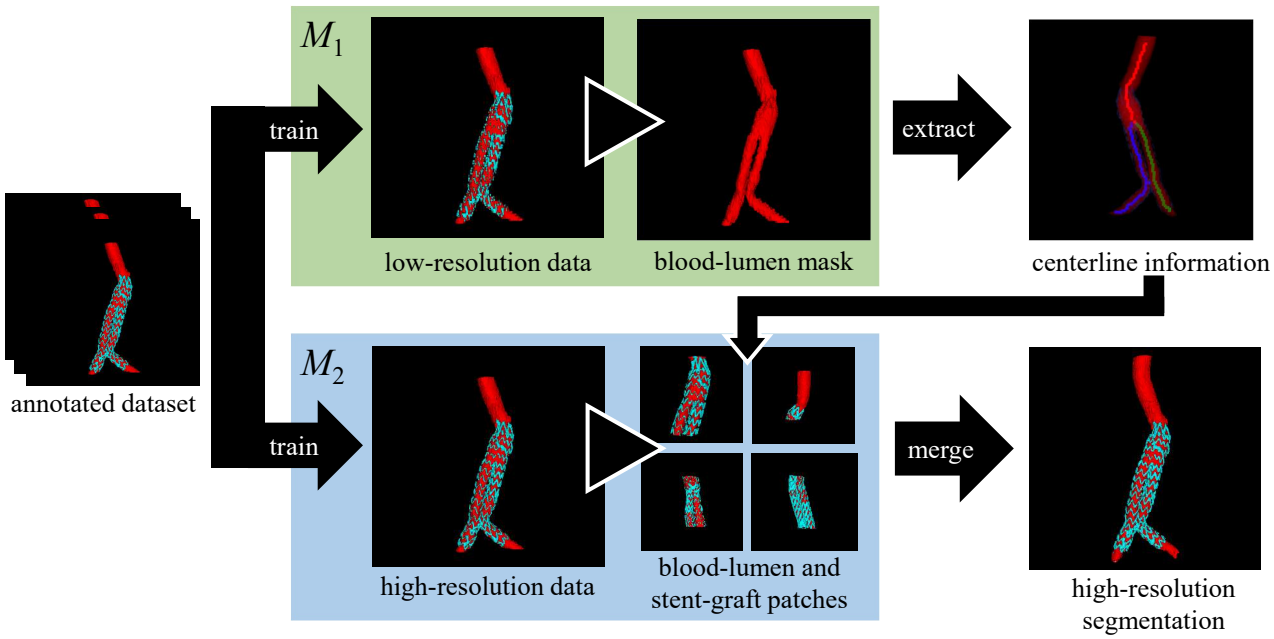
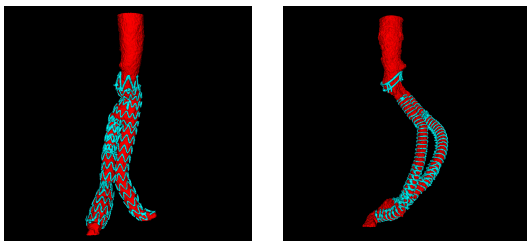


Figure 2. Outline of our method: The top branch shows the centerline extraction step (model M_1) and the bottom branch the patchwise segmentation step (model M_2).

challenges arise due to considerable imaging artifacts caused by the stent-graft wire frame and the distinct boundaries between blood lumen and thrombus. While there are a number of publications on the segmentation of the abdominal aorta, very few have focused on stent segmentation. Klein *et al.* [12] used a graph-based method to create a geometric model of the stent-graft, disregarding the aorta entirely. To the best of our knowledge, there is not a single approach segmenting both structures simultaneously. For the segmentation of the abdominal aorta, traditional approaches include graph-based methods [6, 4, 23] and deformable-models [13, 14] which require user interaction to varying degrees and have predominantly been evaluated on pre-operative scans. A common problem with graph- and deformable-model-based approaches is the introduction of many parameters optimized for the respective dataset, limiting the robustness and applicability of the methods in clinical settings [17]. With the introduction of the convolutional neural network (CNN) the field of medical image analysis changed significantly. Today the U-Net [21] and its 3D equivalent [8] are the most widely models used for medical image segmentation. Both models have been applied to the task of the abdominal aorta segmentation, Zheng *et al.* [26] reporting a Dice similarity coefficient (DSC) of 0.82 for the aneurysm thrombus and Li *et al.* [16] reporting a DSC of 0.92 for the aorta blood lumen. For the segmentation of blood lumen and stent graft wire frame we will therefore likewise rely on the (3D) U-Net architecture. The distinguishing challenge to other segmentation tasks is in our case the fine structure of the stent-graft, with a diameter as small as $0.4\,\mathrm{mm}$ [24], which requires an exceptionally high resolution for accurate reconstruction, pushing the limitations of modern hardware.

3. Dataset

Our dataset consists of 76 abdominal CTA scans of 36 patients treated with EVAR that we received from the Kepler University Hospital Linz. Each scan consists of 155 to 873 axial slices with 512×512 voxels. There are large differences in the resolution with a minimum voxel spacing ranging from $0.404\,\mathrm{mm}$ frontal/sagittal and $0.8\,\mathrm{mm}$ longitudinal



(a) (b)

Figure 3. Examples of two ground truth segmentations: Medtronic Endurant (a) and Anaconda (b). In total the dataset contains 5 different types of stent-grafts.

to $0.977\,\mathrm{mm}$ frontal/sagittal and $3\,\mathrm{mm}$ longitudinal. We used the Active-Contour/Snake-Mode of the Software *ITK-Snap* [25] to semi-automatically create the initial ground truth segmentation of the aortic blood lumen from below the heart to the second iliac bifurcation. The stent-graft segmentation was further added by applying a threshold to a region of interest around the blood lumen. The segmentation was then revised using the Paintbrush Mode of *ITK-Snap*. Figure 3 shows examples of the final ground truth segmentations used for training and validation. The dataset was split into 5-folds using a grouping criterion on the patient number to avoid having multiple scans of the same patients assigned to different folds.

4. Method

We use a two step approach in our segmentation method that is outlined in Figure 2. First we extract the aortic centerlines from a coarse blood-lumen segmentation and subsequently use them to extract high resolution patches along the entire span of the aorta. In the second step we segment the blood lumen and the stent-graft wire frame for each patch and merge the results to a final segmentation. The entire setup is tuned to work with an *NVIDIA GeForce 1080 Ti* (11 GB RAM).

4.1. Centerline Extraction

We use a full-image segmentation model M_1 to create a low resolution segmentation of the aortic blood-lumen. We resample the scans and ground truth to a voxel spacing of $1\,\mathrm{mm}$ frontal/sagittal and $3\,\mathrm{mm}$ longitudinal and crop them to a large region of interest of 192×192 voxels and 128 slices (i.e., a physical extent of $192\,\mathrm{mm}$ frontal/sagittal and $384\,\mathrm{mm}$ longitudinal). The largest connected region of blood lumen voxels in the resulting segmentation is then selected and skeletonized using homotopic thinning [15]. Using the python library *Skan* [20] we extract the centerline graph from the skeletonized images, which is essential for the patchwise segmentation step. Figure 4 outlines the intermediate results of the centerline extraction step and an example patch.

4.2. Patchwise Segmentation

A patchwise segmentation model M_2 is used to segment the aortic blood lumen and the stent-graft wire frame in high resolution patches. We resample the scans and ground truth to a voxel spacing of $0.35\,\mathrm{mm}$ frontal/sagittal and $0.75\,\mathrm{mm}$ longitudinal before extracting patches of size 160×160 voxels and 128 slices. Choosing a high resolution (i.e., small voxel spacing) significantly reduces the amount of distortion introduced by resampling, especially considering the varying voxel spacing in the dataset. However, this results in a rather small physical extent of $56\,\mathrm{mm}$ frontal/sagittal and $96\,\mathrm{mm}$ longitudinal that we seek to use as efficiently as possible by centering the patches at equally distributed locations along the entire centerline graph. In our experiments 100 patches per scan proved more than sufficient to cover the aorta and introduce a significant overlap between the patches. The patches are merged into a final segmentation using a Gaussian-weighted kernel that attenuates voxels at the patch boundaries, where the segmentation results are less reliable.

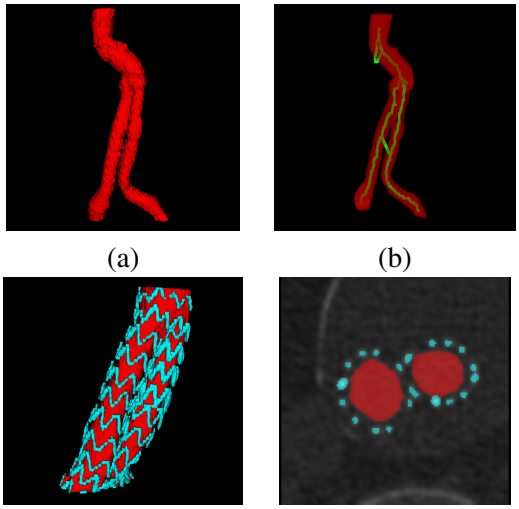


Figure 4. Extracting patches along the segmented aorta lumen: (a) coarse low resolution segmentation of the aortic blood lumen, (b) centerlines approximated via skeletonization of the blood-lumen, (c) ground-truth segmentation for a patch sampled along the centerlines, (d) axial slice of the same patch (scan overlayed with the ground-truth segmentation).

5. Implementation

In this section we discuss the implementation details, i.e., the operations used for preprocessing the dataset, the model architecture and configuration and the training routine.

5.1. Preprocessing

Preprocessing of the dataset consists, in addition to the resampling mentioned in Section 4, of clipping and normalization. As the voxel spacing varies

between the models, the entire preprocessing is done separately for each model. First of all, the dataset is resampled to the respective voxel spacing using a third order B-spline interpolation for the scans and a label-linear interpolation for the ground truth. Next, the intensity values are clipped to the 0.5th and 99.5th percentile over the entire training dataset of the fold. Furthermore, the scans are normalized by subtracting the mean and the standard deviation over the clipped training dataset.

5.2. Architecture

We use the architecture described by Isensee *et al.* [9] and implemented in the Github project 3DUnetCNN [5] as a basis for our experiments. We adjusted the following model parameters: input size, model-depth (number of layers), number of segmentation levels (used for deep supervision) and base-filters (filters in the first convolution kernel). For M_1 (input size of $192 \times 192 \times 128$) we selected a model-depth of 5 with 3 segmentation levels and base-filters set to 8. For M_2 on the other hand (input size of $160 \times 160 \times 128$), we chose an increased model-depth of 6 with 4 segmentation levels and base-filters set to 16. The changes to M_2 were made in order to account for the larger patch size (compared to 128^3 used by Isensee *et al.*) and increase the receptive field of the model. These changes were omitted for M_1, which encompasses a simpler segmentation task, creating only a coarse segmentation of the blood lumen label, while M_2 segments both the blood lumen and the stent-graft wire frame.

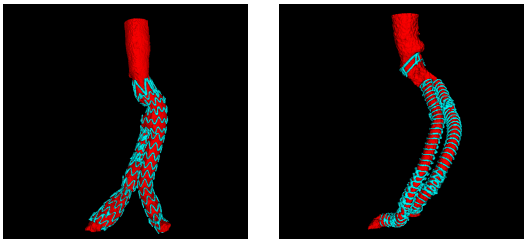
5.3. Training

We trained both models using a weighted multi-class Dice loss [9] in combination with an Adam optimizer. The initial learning rate was set to $\eta_0 = 5 \cdot 10^{-4}$ with a learning rate drop criterion and early stopping after 50 epochs. The training ran for 70 to 120 epochs with 200 training samples per epoch. Due to the 5-fold cross validation used for evaluation, the following statistics are averaged over all folds, where for each fold both models M_1 and M_2 were trained as follows. M_1 was trained first for blood lumen segmentation on the low resolution large regions. The training reached a DSC of 0.978 and 0.898, on average, for the training and validation items, respectively. M_1 was then used to create the blood lumen segmentations for centerline extraction. The resulting centerline graphs were subsequently used during the training of M_2 as the high resolution patches were extracted at random positions along the graph. The average training and validation DSCs for the blood lumen are 0.954 and 0.943, respectively, and 0.843 and 0.841 for the stent-graft.

6. Evaluation

Having trained two models M_1 and M_2 for each fold, we use our method to create high resolution segmentations. Just like during training, M_1 is used to segment the blood lumen used for centerline extraction. The resulting centerline graph is again used to place patches at, however, not randomly but rather at equally distributed positions along the entire span of the graph, as described in Section 4.2. In a post-processing step, the largest connected region of non-background voxels was selected. To compare the results to the ground truth, the segmentations where furthermore resampled to their original voxel-spacing. The last step may be skipped when using the results for further processing rather than evaluation (e.g., mesh generation for blood-flow simulations). Using our method, the cross validation yields an average DSC of 0.961 for the blood lumen and 0.841 for the stent-graft label. Two examples are shown in Figure 5.

To evaluate the effectiveness of our patch extraction method, we further conducted an experiment using only M_2, which was trained using a traditional patch extraction method (see Isensee et al. [10]). Rather than placing the patches along the aorta centerlines, they where placed in a sliding-window fashion, where the patches are aligned in a regular grid of overlapping tiles. The overlap was set to 32 voxels in each dimension (corresponding to $11.2\,\mathrm{mm}$ frontal/sagittal and $24\,\mathrm{mm}$ longitudinal). While this technique was used both during training and inference, the remaining setup (including pre- and post-processing) was left unchanged. We evaluated



(a) (b)

Figure 5. Evaluation results for the two scans shown in Figure 3.

the experiment on the first fold and compared the results to those of our method for the same fold: Although the blood-lumen segmentation is with a DSC of 0.963 slightly better than our method, yielding 0.951 for the same fold, the more complex stent-graft segmentation does not compare well, with a DSC of 0.785 versus our method's score of 0.852.

7. Discussion

The strength of our method is the centerline-guided segmentation method using the aortic centerlines to optimize the patch locations during training and inference. While our method yields better results than a comparable model using a traditional sampling setup, it also reduces the computational cost significantly. For traditional patching using a grid of overlapping tiles (when not allowing the patches to contain regions outside the image) the number of patches calculates as follows:

$$n_{patches} = \prod_{d=1}^{n_d} \lceil \frac{|I|_d - \theta_d}{|P|_d - \theta_d} \rceil \qquad (1)$$

where n_d is the number of dimensions, $|I|_d$ the size of an image, $|P|_d$ the size of a patch and θ_d the overlap in dimension d. For our setup and a chosen overlap of 32 voxels this results in 343 patches on average per scan ($|I| = (990, 990, 678)$, $|P| = (160, 160, 128)$, $\theta_d = (32, 32, 32)$). Increasing this overlap to improve the model's performance quickly raises this number, e.g., an overlap of half the patch size (as used by Isensee *et al.* [10]) would result in 1440 patches on average per scan ($\theta_d = (80, 80, 64)$). The majority of the patches are irrelevant for the result, as they do not intersect with the target structure. By using the aorta centerline information, our method is able to greatly reduce the number of patches, while also optimizing their content for training and inference. As a result, we can target a smaller voxel spacing (which effectively reduces the physical extent of the patches) without the disadvantages of excessive computational costs and poor model performance.

Conclusions

We presented a novel centerline-guided method for fully automated segmentation of the aortic blood-lumen and the stent graft wire frame in abdominal CT-A scans. Using our method, both training and inference can be conducted more efficiently. The evaluated DSC of 0.961 for the blood lumen and 0.841 for the stent graft wire frame suggest results that are suitable for medical analysis. In the future, we plan to use the results of our method for the analysis of risk factors for post-EVAR patients. Futhermore, we plan to extend the use of our method to other medical segmentation tasks.

Acknowledgements

This work was funded by the FFG (Austrian Research Promotion Agency) under the grants 851461 (EndoPredictor), 872604 (MEDUSA) and 867536 (vizARd). This project was supported by the strategic economic and research programme "Innovatives OÖ 2020" of the province of Upper Austria. RISC Software GmbH is a Member of UAR (Upper Austrian Research) Innovation Network.

References

[1] S. Aggarwal, A. Qamar, V. Sharma, and A. Sharma. Abdominal aortic aneurysm: A comprehensive review. *Experimental & Clinical Cardiology*, 16(1):11–15, 2011.

[2] M. Amanuma, R. H. Mohiaddin, M. Hasegawa, A. Heshiki, and D. B. Longmore. Abdominal aorta: characterisation of blood flow and measurement of its regional distribution by cine magnetic resonance phase-shift velocity mapping. *European Radiology*, 2(6):559–564, Dec. 1992.

[3] A. W. Beck, A. Sedrakyan, J. Mao, M. Venermo, R. Faizer, S. Debus, C.-A. Behrendt, S. Scali, M. Altreuther, M. Schermerhorn, B. Beiles, Z. Szeberin, N. Eldrup, G. Danielsson, I. Thomson, P. Wigger, M. Björck, J. L. Cronenwett, and K. Mani. Variations in abdominal aortic aneurysm care: A report from the international consortium of vascular registries. *Circulation*, 134(24):1948–1958, 2016.

[4] J. Egger, B. Freisleben, R. Setser, R. Renapuraar, C. Biermann, and T. O'Donnell. Aorta segmentation for stent simulation. *Computing Research Repository - CORR*, 2011.

[5] D. G. Ellis. 3DUnetCNN. `github.com/ellisdg/3DUnetCNN`, 2018. Accessed: 2020-02-11.

[6] M. Freiman, S. J. Esses, L. Joskowicz, and J. Sosna. An iterative model-constrained graph-cut algorithm for abdominal aortic aneurysm thrombus segmentation. In *2010 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pages 672–675, 2010.

[7] D. P. J. Howard, A. Banerjee, J. F. Fairhead, A. Handa, L. E. Silver, and P. M. Rothwell. Age-specific incidence, risk factors and outcome of acute

abdominal aortic aneurysms in a defined population. *The British Journal of Surgery*, 102(8):907–915, July 2015.

[8] z. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger. 3d U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation. In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*, Lecture Notes in Computer Science, pages 424–432. Springer, Cham, 2016.

[9] F. Isensee, P. Kickingereder, W. Wick, M. Bendszus, and K. H. Maier-Hein. Brain tumor segmentation and radiomics survival prediction: Contribution to the brats 2017 challenge. In A. Crimi, S. Bakas, H. Kuijf, B. Menze, and M. Reyes, editors, *Brainlesion: Glioma, Multiple Sclerosis, Stroke and Traumatic Brain Injuries*, pages 287–297, Cham, 2018. Springer International Publishing.

[10] F. Isensee, J. Petersen, S. A. A. Kohl, P. F. Jäger, and K. H. Maier-Hein. nnU-Net: Breaking the Spell on Successful Medical Image Segmentation. *arXiv:1904.08128 [cs]*, 2019. arXiv: 1904.08128.

[11] K. C. Kent. Abdominal Aortic Aneurysms. *New England Journal of Medicine*, 371(22):2101–2108, Nov. 2014.

[12] A. Klein, J. A. van der Vliet, L. J. Oostveen, Y. Hoogeveen, L. J. S. Kool, W. K. J. Renema, and C. H. Slump. Automatic segmentation of the wire frame of stent grafts from ct data. *Medical image analysis*, 16 1:127–39, 2012.

[13] T. Kovács, P. Cattin, H. Alkadhi, S. Wildermuth, and G. Székely. Automatic Segmentation of the Vessel Lumen from 3d CTA Images of Aortic Dissection. In H. Handels, J. Ehrhardt, A. Horsch, H.-P. Meinzer, and T. Tolxdorff, editors, *Bildverarbeitung für die Medizin 2006*, Informatik aktuell, pages 161–165, Berlin, Heidelberg, 2006. Springer.

[14] F. Lalys, V. Yan, A. Kaladji, A. Lucas, and S. Esneault. Generic thrombus segmentation from pre- and post-operative cta. *International Journal of Computer Assisted Radiology and Surgery*, 12(9):1501–1510, 2017.

[15] T. C. Lee, R. L. Kashyap, and C. N. Chu. Building Skeleton Models via 3-D Medial Surface Axis Thinning Algorithms. *CVGIP: Graphical Models and Image Processing*, 56(6):462–478, 1994.

[16] J. Li, L. Cao, Y. Ge, C. Wang, B. Meng, and W. Guo. Multi-task deep convolutional neural network for the segmentation of type B aortic dissection. *CoRR*, abs/1806.09860, 2018.

[17] K. López-Linares, I. García, A. García-Familiar, I. Macía, and M. Á. G. Ballester. 3d convolutional neural network for abdominal aortic aneurysm segmentation. *CoRR*, abs/1903.00879, 2019.

[18] M. Malina. Reinterventions after open and endovascular AAA repair. *The Journal of Cardiovascular Surgery*, 56(2):257–268, 2015.

[19] S. Moccia, E. De Momi, S. El Hadji, and L. S. Mattos. Blood vessel segmentation algorithms — Review of methods, datasets and evaluation metrics. *Computer Methods and Programs in Biomedicine*, 158:71–91, 2018.

[20] J. Nunez-Iglesias, A. J. Blanch, O. Looker, M. W. Dixon, and L. Tilley. A new Python library to analyse skeleton images confirms malaria parasite remodelling of the red blood cell membrane skeleton. *PeerJ*, 6:e4312, Feb. 2018.

[21] O. Ronneberger, P. Fischer, and T. Brox. *U-Net: Convolutional Networks for Biomedical Image Segmentation*. Springer International Publishing, Cham, 2015.

[22] A. M. Rozenblit, M. Patlas, A. T. Rosenbaum, T. Okhi, F. J. Veith, M. P. Laks, and Z. J. Ricci. Detection of Endoleaks after Endovascular Repair of Abdominal Aortic Aneurysm: Value of Unenhanced and Delayed Helical CT Acquisitions. *Radiology*, 227(2):426–433, May 2003.

[23] T. Siriapisith, W. Kusakunniran, and P. Haddawy. Outer wall segmentation of abdominal aortic aneurysm by variable neighborhood search through intensity and gradient spaces. *Journal of Digital Imaging*, 31, 2018.

[24] R. Winder, Z. Sun, B. Kelly, P. Ellis, and D. Hirst. Abdominal aortic aneurysm and stent graft phantom manufactured by medical rapid prototyping. *Journal of medical engineering & technology*, 26(2):75–78, 2002.

[25] P. A. Yushkevich, Yang Gao, and G. Gerig. ITK-SNAP: An interactive tool for semi-automatic segmentation of multi-modality biomedical images. *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual Conference*, 2016:3342–3345, 2016.

[26] J.-Q. Zheng, X.-Y. Zhou, Q.-B. Li, C. Riga, and G.-Z. Yang. Abdominal Aortic Aneurysm Segmentation with a Small Number of Training Subjects. *arXiv:1804.02943 [cs]*, 2018. arXiv: 1804.02943.

Image Synthesis in $SO(3)$ by Learning Equivariant Feature Spaces

Marco Peer, Stefan Thalhammer, and Markus Vincze
Faculty of Electrical Engineering and Information Technology, TU Wien, 1040 Vienna, Austria
marco.peer@tuwien.ac.at, {thalhammer,vincze}@acin.tuwien.ac.at

Abstract. *Equivariance is a desired property for feature spaces designed to make transformations between samples, such as object views, predictable. Encoding this property in two dimensional feature spaces for 3D transformations is beneficial for tasks such as image synthesis and object pose refinement. We propose the Trilinear Interpolation Layer that applies $SO(3)$ transformations to the bottleneck feature map of an encoder-decoder network. By employing a 3D grid to trilinearly interpolate in the feature map we create models suited for view synthesis with three degrees of rotational freedom. We quantitatively and qualitatively evaluate on image synthesis in $SO(3)$ providing evidence of the suitability of our approach.*

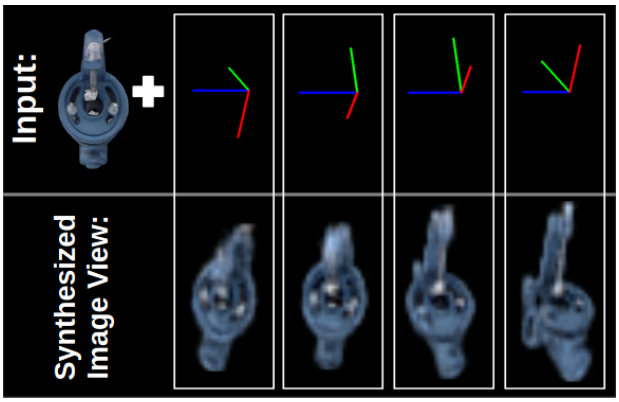


Figure 1: Given an object view and a relative 3D rotation, unseen views are synthesized.

1. Introduction

Invariant feature spaces are agnostic to input transformations in order to help models overcome variations in the data capturing process. Equivariant feature spaces are exploitable with respect to image space transformations, thus more suited for reasoning about changes in image space [9]. As a consequence the property of equivariance is desired for feature spaces that are used for predicting transformations of or in the image space. More precisely, equivariant feature spaces can be exploited to predict unseen views based on known transformations or to estimate relative transformations between two inputs. Feature spaces that correlate an input with a transformed output via observable transformation parameters are desired for applications such as image synthesis or object pose refinement.

In this work we study the equivariance of features spaces of Convolutional Neural Networks (CNN) as motivated by the task of object pose refinement. This motivation arises from recent RGB-based object pose refinement methods that use pairs of images [10, 21]: One image represents the observation of the desired object and the other image usually a rendering of the object in a hypothesized pose. A network is trained to predict the relative transformation between the input pair. We study how to correlate such an image pair, in feature space, in order to achieve predictability of the relative object transformations.

The Spatial Transform Network (STN) [6] provides a mean to learn image space transformations conditioned on the input to produce a transformed output feature map. Studies such as [2, 13] apply a sub part of the STN, known as the Spatial Transformer Layer (STL), to properly align the network's output with its input by applying image space translations. The authors of [19] wrap a projection function around the in- and outputs of the STL in order to make image properties such as lighting and $SO(3)$ transformation in a limited range predictable. Alternatively to their approach, we directly modify the structure of the STL. We extend the STL to enable trilinear interpolation of a feature map in order to interpret transformations in all of $SO(3)$. In the remainder of the paper it is referred to as the Trilinear Interpolation Layer (TIL).

Our contributions are:

- We propose a Trilinear Interpolation Layer suited for creating equivariant feature spaces in $SO(3)$.
- We provide quantitative and qualitative evidence for the advantage of equivariant feature spaces by predicting unseen views in $SO(3)$ of objects from the LineMOD dataset [4].

The remainder of the paper is structured as follows. Section 2 reviews related work. In Section 3 we describe our approach. Section 4 presents our experimental results. Finally, Section 5 concludes the paper.

2. Related Work

Object pose refiners rely on the availability of prior stages to produce pose hypotheses [7, 10, 12, 16, 18, 20, 21]. When depth data is available, the Iterative-Closest-Point algorithm (ICP) can be used to refine initial pose estimates [18, 7, 20]. Recent RGB-based approaches do not rely on the availability of depth data for pose refinement [7, 10, 12, 16, 21]. CNN-based object pose refinement architectures such as [10, 12, 21] pass two input images to the network in order to estimate the relative rotation between these. These images are an observation of the object in the desired pose and a rendering of the prediction. In [10] the authors base their network architecture on an approach for optical flow estimation [1] and predict optical flow, mask and relative pose deviation in $SE(3)$. The authors of [21] use a similar approach with two encoders, one per input image. The encoders' outputs are subtracted and further encoded to predict the refined pose in $SE(3)$. We present a concept suitable for enhancing such methods by guiding the network to learn an equivariant feature space.

The STN introduced by [6] is widely used for feature and image space transformation [2, 13, 14, 15, 19]. It consists of the combination of a localization network, a grid generator and a sampler.. The authors of [2] apply STL to properly align the features to their inputs. In [13] the authors predict deep heatmaps from randomly sampled object patches to predict poses under occlusion. They apply the STL to upsample their predictions. In [14, 15] an analog of the localization network is used to produce feature maps invariant to input transformations. The authors

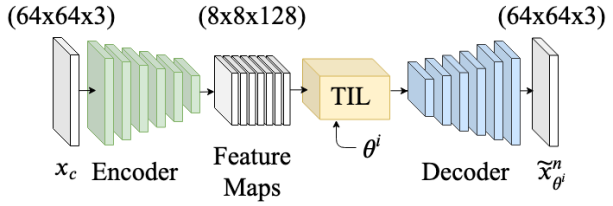


Figure 2: Encoder-decoder architecture for image synthesis.

of [11] leverage on the methodology of STN to generate realistic looking images from the intersection of the natural image and geometric manifold, using an adapted Generative Adversarial Network. Conversely to these approaches we modify the STL component of STN to enable $SO(3)$ transformations of input feature maps with spatial dimension.

3. Approach

This section presents our approach for learning equivarient features in $SO(3)$ in order to synthesize images from unseen viewpoints. We first give a problem definition, then describe the Trilinear Interpolation Layer. Finally, we outline how the TIL is used in an encoder-decoder architecture for image synthesis.

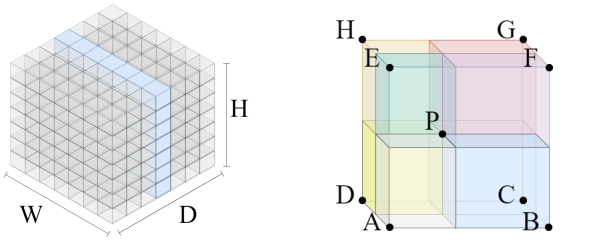
3.1. Problem Statement

Let $X = \left\{x_c, \left(\tilde{x}^0_{\theta^0}, ..., \tilde{x}^n_{\theta^i}\right)\right\}$ be a set of training examples where x_c refers to the projection $\prod$ of object o_c, in its canonical pose, to the image space I. The set of $\tilde{x}^n_{\theta^i}$ are the projections of transformed objects o_{θ^i} where θ^i represent the transformation in $SO(3)$ for the projection into I. Our goal is to learn the inverse of the mapping function $\prod^{-1}$ in order to produce transformed images. In other words, to learn $\tilde{x}^n_{\theta^i} = \prod\left[\prod^{-1}(x_c), \theta^i\right]$ given an image of the object in its canonical pose and transformation parameters.

In order to model the inversion of the mapping function $\prod$, we utilize a CNN due to their power to encode statistical relationships from visual data into feature spaces [8]. To provide information regarding relative transformations θ^i in $SO(3)$ between pairs of images to our model, we modify the STL of [6]. An overview of the encoder-decoder architecture for image synthesis using the modified STL is presented in Figure 2.

3.2. Trilinear Interpolation

The STL [6] allows $SE(2)$ transformations to be applied to feature maps. This works well in image



(a) Three dimensional grid. The feature map are centered in D on initialization.

(b) Interpolation scheme of P using adjacent grid cell values.

Figure 3: Trilinear Interpolation Layer (TIL) components.

space, however, requires adaptation for the $SO(3)$ domain [19]. The STL is composed of a grid generator and a sampler.

The grid generator is modified by adding a depth dimension D. The input feature map of space $\mathbb{R}^{H\times W\times C}$ thus becomes $\mathbb{R}^{H\times W\times D\times C}$, where H, W and C are the height, width and number of channels, respectively. The feature map is centered along D as shown in Figure 3a. The sampler of the STL bilinearly interpolates between corner points using the corresponding areas. For volumes, this scheme is unsuitable. Therefore, trilinear interpolation is used instead, as shown in Figure 3b. Feature maps are interpolated channel-wise and projected back to 2D by averaging along the depth dimension. In order to guarantee proper interpolation in 3D, H and W must be greater than 1. The proposed modification enables transformations in $SO(3)$ and only affects non-trainable layers. Thus, the additional computational overhead compared to STL is negligible.

Since averaging over D is used for projecting the grid back to 2D no feature map scaling can be applied while sampling. Thus, modifying the trilinear interpolation by allowing scaling along depth would enable transformations in $SE(3)$, thus yielding full 6DoF. However, this is out of the scope in this paper.

3.3. Network Architecture

The network in Figure 2 is an encoder-decoder architecture. The encoder consists of a truncated ResNet18 [3], pretrained on ImageNet [17], for feature encoding. ResNet18 consists of five stages. In order to preserve a larger spatial image dimension we remove the fourth and the fifth stage and take the outputs of the last Rectified Linear Unit (ReLU) of stage three. The final output is a tensor of size 8×8 with 128 feature maps.

The encoded image $\prod^{-1}(x_c)$ as well as the transformation parameters θ^i are passed to the TIL. Feature maps are trilinearly interpolated to produce the mapping of the encoded transformed image $\tilde{x}^n_{\theta^i}$. The transformed encoding is forwarded to the decoder stage of the network.

The design of the decoder is rather ad-hoc to show that the TIL is not restricted to a certain architecture. A transposed convolution with ReLU activation is followed by stacks of deconvolution layers with ReLU activation and upsampling layers. These stacks are repeated two times and a final transposed convolution layer with linear activation is added. Feature channels are reduced gradually. Kernel sizes of the transposed convolutional layers are $5-3-3-5$ and upsampling kernel sizes of 3×3 are used. All strides are set to 1. The output of the decoder is an image of size 64×64.

In each training iteration, the deviation of $\tilde{x}_\theta$ to x_θ is minimized. The loss function to be optimized is $l2$. The network is trained to correlate objects views with its corresponding transformation in $SO(3)$ in the camera frame. Consequently, a feature space is created that enables to synthesize views not included in the training set.

4. Experiments

This section presents experiments for image synthesis of unseen views of household objects with little texture. These experiments show that the extension of an encoder-decoder network with the proposed TIL reconstructs objects views in $SO(3)$. In addition, the method can also reconstruct views in regions of $SO(3)$ where no data was provided to the network during training.

4.1. Dataset

Our experiments are conducted on a subset of the LineMOD dataset [4]. We use the object models of *Benchvise*, *Cat*, *Glue*, *Camera* and *Lamp*. These objects represent elongated and asymmetric shapes as well as complex shapes with self occlusion. With this subset we cover the representative challenges when synthesizing views for objects.

4.2. Dataset Creation

Dataset images are rendered using the renderer provided by [5]. For our purposes, the RGB images are scaled to 64×64 pixels. To each object's canonical pose, $45°$ are added to elevation in order to only train on views of the upper hemisphere of the object.

metric	latent space	loss			
		l1	*l2*	DSSIM	DSSIM + *l1* ($\delta = 0.85$)
l1	2x2x512	0.03 ± 4.9e-04	0.03 ± 4.2e-04	0.028 ± 4.0e-04	0.031 ± 4.3e-04
l2		0.096 ± 2.1e-05	0.093 ± 1.6e-02	0.093 ± 1.8e-03	0.1 ± 1.2e-03
DSSIM		0.102 ± 3.5e-03	0.105 ± 2.6e-03	0.09 ± 3.0e-03	0.1 ± 3.4e-03
l1	4x4x256	0.018 ± 3.5e-04	0.02 ± 3.6e-04	0.0243 ± 4.0e-04	0.018 ± 3.2e-04
l2		0.065 ± 1.8e-05	0.0064 ± 1.7e-05	0.086 ± 2.5e-06	0.063 ± 1.7e-05
DSSIM		0.061 ± 3.0e-05	0.067 ± 3.6e-05	0.07 ± 2.8e-05	0.059 ± 3.2e-05
l1	8x8x128	**0.016 ± 2.6e-04**	0.017 ± 2.6e-04	0.017 ± 25e-05	0.017 ± 2.4e-04
l2		0.06 ± 1.6e-03	**0.057 ± 1.7e-03**	0.06 ± 1.7e-03	0.066 ± 1.6e-03
DSSIM		0.055 ± 3.0e-03	0.06 ± 3.0e-03	**0.053 ± 2.9e-03**	0.055 ± 2.6e-03

Table 1: Performance study for latent spatial dimension and loss function. We present the error and variance, averaged over all objects, using *l1*, *l2* and DSSIM respectively.

Based on the newly defined canonical pose, images are rendered in a range of -43° to +43° azimuth and elevation. This is similar to [19] but with approximately three times the range in azimuth angle.

For training, only views in a range of -37° to +37° azimuth and elevation are used. Of these 950 images, 43 images are exclusively used for testing. The selected samples are distributed uniformly in the viewing cone. An additional 59 images are included in the test set. These are in an angle range of negative and positive 37° to 43° azimuth and elevation. Thus, views in a range that are not shown to the network during training.

4.3. Training Protocol

For training we use the Adam optimizer with the learning rate set to 10^{-3}. A batch size of 1 is used. We train 40 epochs per object for quantitative ablation studies. After 30 epochs, the learning rate is decreased by one magnitude. Qualitative evaluation is presented after 40 epochs of training. During training, Gaussian blur with uniformly sampled $\sigma = [0.0, 1.5]$ is used as online augmentation.

4.4. Hyperparameter Studies

We study the choice of loss function used for optimization and the optimal size of the bottleneck feature maps. Table 1 presents results averaging over the test sets of all five objects. Presented are the Mean Absolute Error (MAE), Root Mean Squared Error (RMSE) and Structural Similarity Index (SSIM) as well as their corresponding variances.

The loss functions compared are $l1$, $l2$, Structural Disimilarity (DSSIM) and a combination of $l1$ and DSSIM as used by [19], where δ is the weighting parameter. The bottleneck tensor size is adjusted by truncating ResNet18. For a dimension of $4 \times 4 \times 256$ we use the outputs of the fourth and upsample using three stacks of transposed convolutions plus upsampling layers. For $2 \times 2 \times 512$ we use four stacks starting with a 5×5 transposed convolution.

Tensor size	2x2x512	4x4x256	8x8x128
parameters	13,330,508	3,753,804	1,881,932

Table 2: Network parameters per bottleneck tensor size.

Quantitative evaluation shows that the metric used for evaluating the reconstruction quality correlates with the loss function used, which is to be expected. Using $l2$ is reasonable. However, when synthesizing views for a specific application more carefully choosing the loss function will be obligatory. Surprisingly, a bottleneck tensor size of $8 \times 8 \times 128$ leads to image synthesis with the lowest error even though this network has far fewer parameters than the other spatial dimensions (see Table 2). This leads to the conclusion that bigger spatial dimensions are more important for synthesizing views than network depth. Based on the chosen hyperparameters we further present experiments for synthesizing views.

4.5. Studies on View Synthesis

Studies are presented to illustrate that the proposed formulation generates feature spaces suited for view synthesis in $SO(3)$. Figure 4 shows views synthesized from unseen transformations during training time. Additionally, we present view predictions outside of the training range. Views inside the training range are reconstructed with sufficient quality to visually verify the expected object orientations. Despite the reconstruction quality being poor for

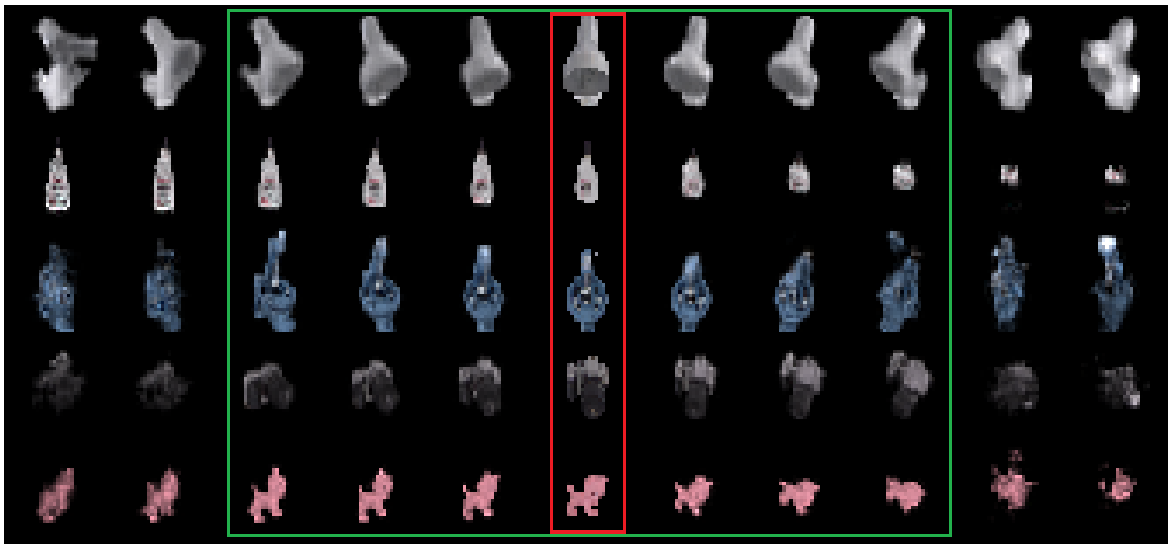


Figure 4: View synthesis from $SO(3)$ transformations unseen during training time. First row: reconstructed *Lamp* with varying azimuth from -43° to 43°. Second row: reconstructed *Glue* with elevation variation from -43° to 43°. Row three to five: objects *Benchvise, Camera, Cat* reconstructed with azimuth/elevation range from (-43°,-43°) to (43°,43°). Object poses outside the green box are samples out of training distribution. Centered images, in the red box, mark the canonical poses.

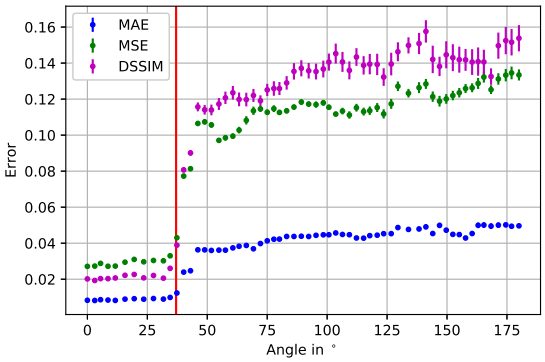


Figure 5: Error values and its variance over azimuth angle. The network was trained on its corresponding loss function with a spatial bottleneck dimension of $8 \times 8 \times 128$. The vertical line shows the training set range.

some of the synthesized views outside of the training range, it is visible that views can be predicted properly based on $SO(3)$ transformations.

Figure 5 provides reconstruction error and variance over an extended azimuth and elevation angle range of [0, 180°]. The results in the figure are averaged over all objects. The training dataset contains images with azimuth angles up to 37°. A sharp rise in error and variance is observed at azimuth angle of approximately 45°. For angles above this value, error and variance increase rapidly. As such, the network cannot properly reconstruct these views.

These results show that our formulation for creating equivariant feature spaces has the desired property to correlate spatial transformations with 2D views of the transformed object. Thus, the proposed Trilinear interpolation layer guides the network towards learning an equivariant feature space in $SO(3)$.

5. Conclusion

We extend recent work for learning equivariant feature spaces for synthesizing object views in $SO(3)$. The proposed extension of the Spatial Transform Network [6], that we call the Trilinear interpolation Layer, applies $SO(3)$ transformations to feature maps from 2D data. Validity of the approach is provided by training a simple encoder-decoder network architecture. Our experiments show that our formulation not only enables the prediction of views unseen during training time but also in a small range outside.

The current formulation enables control for 5DoF, $SO(3)$ and translations in image space. Future work will tackle adapting the proposed layer to create object view synthesis in all of $SE(3)$. We then plan to integrate this in a pose refinement strategy to improve object pose estimation.

ACKNOWLEDGMENT

This work has been supported by the Austrian Research Promotion Agency in the program Production of the Future funded project MMAssist_II (FFG No. 858623), the Austrian Ministry for Transport, Innovation and Technology (bmvit) and the Austrian Science Foundation (FWF) under grant agreement No. I3969-N30 (InDex).

References

[1] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2758–2766, 2015.

[2] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[3] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[4] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, G. Bradski, K. Konolige, and N. Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Proceedings of Asian conference on computer vision*, pages 548–562, 2012.

[5] T. Hodan, F. Michel, E. Brachmann, W. Kehl, A. GlentBuch, D. Kraft, B. Drost, J. Vidal, S. Ihrke, X. Zabulis, et al. Bop: Benchmark for 6d object pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 19–34, 2018.

[6] M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. In *Advances in neural information processing systems*, pages 2017–2025, 2015.

[7] W. Kehl, F. Manhardt, F. Tombari, S. Ilic, and N. Navab. SSD-6D: making rgb-based 3d detection and 6d pose estimation great again. *CoRR*, abs/1711.10006, 2017.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[9] K. Lenc and A. Vedaldi. Understanding image representations by measuring their equivariance and equivalence. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 991–999, 2015.

[10] Y. Li, G. Wang, X. Ji, Y. Xiang, and D. Fox. Deepim: Deep iterative matching for 6d pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 683–698, 2018.

[11] C.-H. Lin, E. Yumer, O. Wang, E. Shechtman, and S. Lucey. St-gan: Spatial transformer generative adversarial networks for image compositing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9455–9464, 2018.

[12] F. Manhardt, W. Kehl, N. Navab, and F. Tombari. Deep model-based 6d pose refinement in rgb. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 800–815, 2018.

[13] M. Oberweger, M. Rad, and V. Lepetit. Making deep heatmaps robust to partial occlusions for 3d object pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 119–134, 2018.

[14] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

[15] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in neural information processing systems*, pages 5099–5108, 2017.

[16] M. Rad and V. Lepetit. Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3828–3836, 2017.

[17] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115:211–252, 2015.

[18] M. Sundermeyer, Z.-C. Marton, M. Durner, M. Brucker, and R. Triebel. Implicit 3d orientation learning for 6d object detection from rgb images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 699–715, 2018.

[19] D. E. Worrall, S. J. Garbin, D. Turmukhambetov, and G. J. Brostow. Interpretable transformations with encoder-decoder networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5726–5735, 2017.

[20] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. 2018.

[21] S. Zakharov, I. Shugurov, and S. Ilic. Dpod: 6d pose object detector and refiner. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1941–1950, 2019.

Frame Border Detection for Digitized Historical Footage

Helm Daniel, Pointner Bernhard, Kampel Martin
TU Wien, Institute for Visual Computing and Human Centered Technology
{daniel.helm,martin.kampel}@tuwien.ac.at, bernhard.pointner@student.tuwien.ac.at

Abstract. *Automatic video analysis of digitized historical analog films is influenced by video quality, composition and scan artifacts called overscanning. This paper provides a first pipeline to crop the main frame window by detecting Sprocket-Holes and interpreting the geometric hole layout to distinguish between two different film reel types (16mm and 9.5mm). Therefore, an heuristic approach based on histogram features is explored. Finally, our results demonstrate a first baseline for future research.*

1. Introduction

In the age of digitization analog film collections are digitized by using modern technologies and processes[1]. During these processes the frame content as well as the area around the exposed frame is scanned. This area includes black borders of the film reel, *Sprocket-Holes (SH)* or parts of the next or previous frames. This effect is called *overscanning* and is needed to ensure preservation of significant information (see Fig.1-a). Furthermore, it is a fundamental procedure for sustainable film digitization and archival. However, for developing automatic video analysis tools of scanned historical analog films, this additional information is undesirable and can influence the performance of those systems [1, 3, 4]. The project *Visual History of the Holocaust (VHH)*[2] has been funded in order to digitize analog media collections related to the liberation phase of the Nazi concentration camps. These collections are used for further explorations on automatic video content analysis. However, they do not include annotated metadata such as the film reel type or masked overscan areas. Therefore, automatic mechanisms for detecting and removing overscans in film reels such as 16mm or 9.5mm (see Fig.1-b) can be used to provide more efficient ways for exploring analog films.

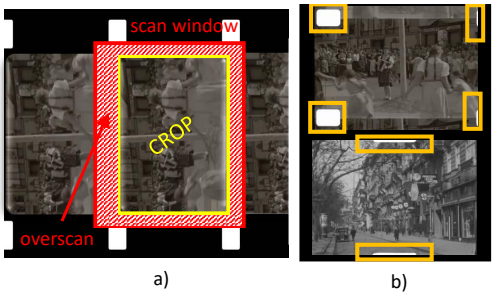


Figure 1. (a) Demonstration of *overscanning*, (b) real world examples of a 16mm (top) and 9.5mm (bottom) film reels.

Mühling et al. [2] and Zeppelzauer et al. [4] explore the challenges of cinematographic techniques in historical videos. However, to our best knowledge no comparable scientific investigation on automatically removing overscan information by detecting SH has been published in the last decade. This paper proposes a first Frame-Border-Detection (FBD) approach to remove overscan areas in scanned analog frames by detecting SHs as well as interpreting the hole geometry and layout. This information is used to classify two different film reel types (16mm and 9.5mm). Moreover, the hole positions are used to extract the final frame window using traditional computer vision techniques.

2. Methodology

We propose a multi-stage pipeline split into four main blocks: *Threshold-Filtering (THF)*, *Connected-Component-Labelling (CCL)*, *Calculating-Crop-Window (CCW)* and *Reel-Type-Classifier (RTC)*. The original input frame is first converted into a grayscale image. In the THF-stage, the input image is thresholded to get a binary mask. The threshold T_{h} is calculated for each input frame dynamically by analyzing the fields 1-6 visualized in Figure

[1] https://dft-film.com/products/archive-challenges-and-solutions.html - last visit: 2020/02/08

[2] https://www.vhh-project.eu/en/ - last visit: 2020/02/08

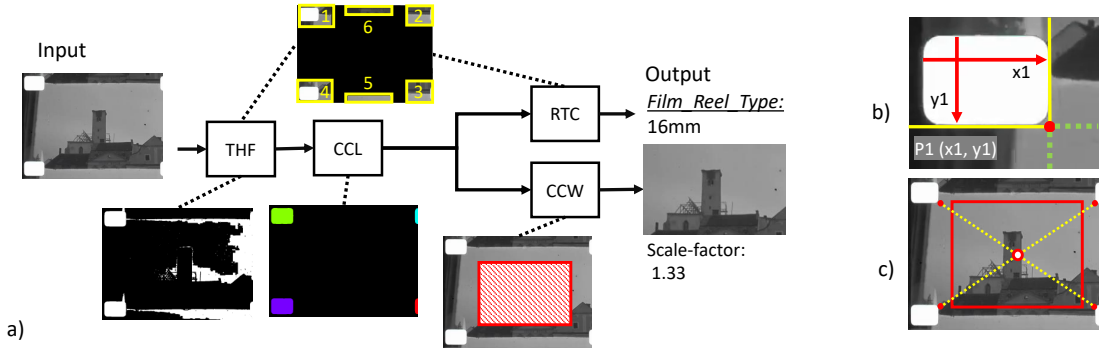


Figure 2. Illustration of (a) the pipeline, (b) the generation of the corner point and (c) calculation of the final crop window.

Exp.	P	R	Acc	mIoU@0.95	IoU@0.70
16mm-fixed	0.95	0.94	0.88	-	-
9.5mm-fixed	0.94	0.68	0.82	-	-
16mm-dyn.	0.96	0.84	0.90	-	-
9.5mm-dyn.	0.97	0.64	0.81	-	-
overall-fixed	0.948	0.812	0.85	0.747	0.867
overall-dyn.	**0.962**	**0.74**	**0.86**	**0.763**	**0.895**

Table 1. Precision (P), Recall (R) and Accuracy (Acc) on the test set - classification of the reel-types 16mm and 9.5mm. mean Intersection over Union scores at thresholds: 0.95 and 0.7.

2-a. After a filtering process, the mask is used in the CCL-stage for labeling all detected potential SH. This step includes a further filtering process to remove outliers. Finally, this step is the base for the CCW- and RTC-stage. In CCW the corner points are calculated as demonstrated in Figure 2-b. The center of the resulting square is used as reference point for the final crop window which is defined with a configurable scale factor (e.g 1.33) to get the correct scaled frame crop related to the original film reel such as 960x720 pixels (see Fig.2-c). In the RTC-stage, our pipeline is able to classify the input frame into the reel types: 16mm and 9.5mm. Therefore, the locations of the labeled holes are analyzed in the masked input image. SHs in the fields 5 and 6 (see Fig. 2-a) are related to the 9.5mm film reel whereas the other ones identify the 16mm reels.

3. Results & Conclusion

The evaluation of our pipeline is based on a self-generated dataset including 100 labeled frames randomly selected out of 10 videos related to the National-Socialism[3]. The dataset contains 50 annotated frames for each class: 16mm and 9.5mm reels. The metric mean Intersection over Union (mIoU) is used for evaluating the predicted locations. Precision and Recall are utilized to evaluate the classification of the two reel types. For the evaluation, two experiments have been conducted: a fixed and dynamic T_{h}. The results show that the mIoU scores significantly depending on the THF process. Historical film frames include damaged and under-/overexposed areas which make the selection of an optimal T_{h} challenging. Furthermore, SHs are not on the same positions in each frame due to the movements and the varying speed of the film reel during the scan process. The results are summarized in Table 1. We provide a first baseline for further research. However, optimizing our pipeline as well as using Deep Learning-based methods are planned to improve detection and classification results.

[3]http://efilms.ushmm.org/ - last visit: 2020/02/11

Acknowledgments

The project VHH has received funding from the EU's H2020 research program (Grant No. 822670).

References

[1] D. Helm and M. Kampel. Video Shot Analysis for Digital Curation and Preservation of Historical Films. In S. Rizvic and K. Rodriguez Echavarria, editors, *Eurographics Workshop on Graphics and Cultural Heritage*. The Eurographics Association, 2019.

[2] M. Mühling, M. Meister, N. Korfhage, J. Wehling, A. Hörth, R. Ewerth, and B. Freisleben. Content-based video retrieval in historical collections of the german broadcasting archive. *Int. J. Digit. Libr.*, 20(2):167–183, June 2019.

[3] M. Seidl, M. Zeppelzauer, D. Mitrović, and C. Breiteneder. Gradual transition detection in historic film material - a systematic study. *J. Comput. Cult. Herit.*, 4(3):10:1–10:18, 2011.

[4] M. Zeppelzauer, D. Mitrovic, and C. Breiteneder. Archive film material - a novel challenge for automated film analysis. *The Frames Cinema Journal*, 1(1), 2012.

Highly Accurate Binary Image Segmentation for Cars

Thomas Heitzinger, Martin Kampel
Computer Vision Lab, TU Wien, Austria
{thomas.heitzinger,martin.kampel}@tuwien.ac.at

Abstract. *We study methods for the generation of highly accurate binary segmentation masks with application to images of cars. The goal is the automated separation of cars from their background. A fully convolutional network (FCN) based on the U-Net architecture is trained on a private dataset consisting of over* 7000 *samples. The main contributions of the paper include a series of modification to common loss functions as well as the introduction of a novel Gradient Loss that outperforms standard approaches. In a specialized postprocessing step the generated masks are further refined to better match the inherent curvature bias typically found in the outline of cars. In direct comparison to previous implementations our method reduces the segmentation error measured by the Jaccard index by over* 65%.

1. Introduction

A majority of buyers and sellers of cars choose to use online platforms. The quality of pictures on such platforms has a considerable impact on a buyers likelihood to purchase and thus leads to a demand for visually appealing images. For most sellers it is financially infeasible to take professional photographs and it has instead become common practice to digitally edit them. A binary segmentation mask is created that segments the image into foreground (the vehicle) and background. This mask is used to either alter (e.g. blur) or entirely replace the background with an artificial scene. Due to the significant demand for high quality segmentation masks dedicated businesses offering this service have emerged. As each photograph is edited by hand, the total time until the segmentation mask is available to the dealership lies between one and two days. The delay in time generates non-negligible costs. Based on novel deep learning techniques that have advanced the state-of-the-art in recent years we study methods for the fully automated generation of segmentation masks with focus on the maximization of accuracy. This paper aims to improve the state of the CarCutter[1] service.

2. Related Work

The first application of convolutional networks to semantic segmentation with per-pixel prediction was made possible by the introduction of fully convolutional networks (FCN) [13]. Previously segmentation solutions repurposed convolutional network architectures [12, 4] intended either for classification or object detection and always included fully connected layers. These adaptations come with drawbacks on either speed or accuracy. By reinterpreting fully connected layers in classification networks as convolutional layers that cover the entire input region the network architecture is made independent of the dimensions of the input image. Instead of a class probability vector the reinterpreted network outputs a coarse heatmap for each class. In order to obtain predictions at the pixel level the coarse semantic information of deeper levels is repeatedly upsampled and added to the activations of shallower feature maps. This innovation was quickly expanded on and led to development of the U-Net architecture [11]. It introduces a symmetric encoder-decoder format consisting of a contracting encoder component and an expanding decoder component. This setup is chosen with the intention of learning a comparatively low dimensional image representation in the narrow region of the network (referred to as the *bottleneck*) that captures global context while at the same time dramatically reducing the number of learned parameters. Skip connections efficiently pass shallow encoder features with high localization accuracy to deep decoder layers that are rich in semantic information. Variations on networks of this

[1] https://www.car-cutter.com/ (accessed February 24, 2020)

type are often focused on the decoder component, while the standard approach for the encoder component is the repurposing of the convolutional stage of known, well performing networks, such as VGG-16 [14]. The variations in the decoder component essentially explore the trade-off between low memory requirements (and fast inference) and high accuracy. Architectures such as [16] also investigate the benefits of an additional ResNet [5] based refinement stage. Benchmarks show [2, 10] that almost all state-of-the-art solutions for a variety of image segmentation tasks are based on the U-Net architecture. It is also chosen by well performing entries [6, 15] to the Kaggle Carvana Image Masking challenge. TernausNet [6] was part of the winning entry in the challenge and uses a pretrained encoder based on VGG-11 [14] while [15] placed in the top 4% using an ensemble of five network with a pretrained ResNet-50 [5] encoder.

3. Dataset

Training was done on a private dataset consisting of 7614 pairs of RGB-images and binary segmentation masks. Some images contain additional cars in the background that are smaller by area. In these cases the solution is expected to only segment the main vehicle. The dataset exhibits a bias towards German car brands such as Volkswagen, BMW and Mercedes and contains a disproportionate amount of images with cars higher-than-average in cost. During preprocessing all images are resized to a resolution of $800\text{px} \times 600\text{px}$. Data augmentation is used to boost the available training data.

To our knowledge, the most closely related dataset is tied to the Kaggle Carvana Image Masking challenge[2]. The goal of this challenge is identical to ours. Its dataset contains roughly $100\,000$ image/mask pairs with resolution $1920\text{px} \times 1080\text{px}$. Compared to our dataset the samples are more uniform. Each picture contains exactly one vehicle which is placed in a fixed position and all photographs are taken by the same stationary cameras under identical lighting conditions. The winning entry of this challenge achieved a Jaccard index of 0.9947 which we consider to be an upper bound to the score achievable on our dataset.

[2]https://www.kaggle.com/c/carvana-image-masking-challenge (accessed February 21, 2020)

4. Methods

Segmentation is performed with a fully convolutional neural network of the U-Net architecture. Its implementation is similar to [16], with a pretrained convolutional stage of a VGG-16 network with batch normalization for the encoder and an additional ResNet-style refinement block after the decoder. Segmentation quality is evaluated using the Jaccard index which is the de facto standard metric for image segmentation methods:

$$\mathcal{M}_{\mathrm{J}}(P, T) \coloneqq \frac{|P \cap T|}{|P \cup T|}. \tag{1}$$

In our context T and P are subsets of target (ground truth) and predicted pixels in a segmentation mask. Images x, target masks t and predicted masks p are assumed to be non-binary with height N, width M and values in the range $[0, 1]$.

We study training with (modifications of) the loss functions Mean Squared Error $\mathcal{L}_{\mathrm{MSE}}$ and Binary Cross-Entropy $\mathcal{L}_{\mathrm{BCE}}$ as well as the Dice Loss [9] $\mathcal{L}_{\mathrm{DSC}}$ which is defined as

$$\mathcal{L}_{\mathrm{DSC}}(p, t) \coloneqq 1 - \frac{\epsilon + 2\sum_{(i,j)\in D} p_{ij} t_{ij}}{\epsilon + \sum_{(i,j)\in D} p_{ij} + t_{ij}}, \tag{2}$$

and is related to the Jaccard index. Here $D = \{1 \,..\, N\} \times \{1 \,..\, M\}$ is the domain of the segmentation masks and $\epsilon \ll NM$ is a small scalar regularization term.

4.1. Weighting Schemes

We propose modifications that improve upon the standard losses Mean Squared Error and Binary Cross-Entropy. The main idea is that not all areas of an image are equally important or equally difficult to segment. Loss functions that are the sum or mean of pixelwise losses can be modified to assign weights to each pixel in order to adjust for this inhomogeneity. We can use a map w of real weights with shape equal to t and p and define a modified version of Mean Squared Error as:

$$\mathcal{L}_{\mathrm{MSE}}(p, t) \coloneqq \frac{1}{NM} \sum_{(i,j)\in D} w_{ij} \left(p_{ij} - t_{ij}\right)^2. \tag{3}$$

An analogous modification can be made to Binary Cross-Entropy.

Notation The notation $\nabla_\sigma y$ expresses the convolution of a stack of feature maps y with the gradient of a two-dimensional Gaussian density with mean vector $(0,0)^{\mathrm{T}}$ and covariance matrix σI, where I is the identity matrix. In practice it is a convolution $\nabla_\sigma y = y * G_\sigma^\nabla$ with a kernel G_σ^∇ that is normalized and has shape $2 \times C \times C$ with $C \sim 4\sigma$. The operation doubles the channels of the tensor y. To ensure fast computation the convolution is implemented as a convolutional layer with frozen weights.

Median Frequency Balancing A simple and popular [3, 1] weighting scheme is *Median Frequency Balancing* (MFB). Each class (foreground and background in our binary setting) is assigned a weight to compensate for imbalance in the frequency of occurrence. The weights can either be computed individually for each sample or once for the entire dataset. In the individual case a foreground/background weight pair (w_f, w_b) for a target mask t is given by

$$w_f = \frac{N}{2\sum_{(i,j)\in D} t_{ij}} \quad \text{and} \quad w_b = \frac{N}{2\sum_{(i,j)\in D}(1-t_{ij})}. \tag{4}$$

An example of such a weight map is shown in the second column of Figure 1. If a single weight pair for the entire dataset is preferred then it is computed as the mean of all sample weights.

Boundary Proximity The separating boundary between foreground and background is the only area where the segmentation mask is non-constant. Consequently it is also the area where masks generated by neural networks exhibit the largest mistakes. In this approach pixels in close vicinity to the boundary are assigned larger weights. Such a method is already suggested by the authors of the original U-Net architecture [11], although with a less general approach. We calculate a weight map based on a pixel's distance to the separation boundary using convolution based edge detection. A large gradient in a segmentation mask indicates the presence of an edge. Based on this we define the weight map

$$w_{ij} = 1 + c\|(\nabla_\sigma t)_{ij}\|_2^2 .$$

A map of this type is shown in the third column of Figure 1. The parameter c is a scaling constant and is set to 5.

Gradient Ratio The typical location of segmentation errors can be characterized more concretely. Photographs are often taken in poor lighting conditions or with cheap camera equipment resulting in over- or underexposed areas. Common occurrences are bright reflections in a vehicles roof or dark shadows around its wheelbase (see Figure 2). Both scenarios can obscure the precise transition point between foreground and background. At the data level we are confronted with image patches that are either nearly entirely white or nearly entirely black, while the same patch in the ground truth segmentation mask contains a binary transition. Motivated by this observation we claim that the ratio between change in the mask and change in the corresponding image is a measure for prediction difficulty and use it to define a new weight map. Again we employ discrete gradients:

$$w_{ij} = 1 + c\,\frac{\|(\nabla_\sigma t)_{ij}\|_2^2}{\|(\nabla_\sigma x)_{ij}\|_2^2 + \epsilon}.$$

As previous the parameter c is a constant which is set to 0.1 and ϵ is a small regularizing constant with $\epsilon \ll NM$. The result of the convolution $\nabla_\sigma x$ is a stack of six feature maps, one for each combination of the three image channels and the two partial derivatives in the gradient. A weight map of this type is shown in the fourth column of Figure 1.

Comparative Results In Table 1 we show results for the pixelwise losses Mean Squared Error and Binary Cross-Entropy, first in their default state and then with the addition of one or more weighting extensions. To us a pixelwise loss is a function that sums over the losses of individual pixels. Consequently when the gradient is computed during backpropagation all terms except the ones belonging to the individual pixels vanish. We can argue that in such a loss function no pixel is *ignored* or treated lesser.

In direct comparison Binary Cross-Entropy outperforms Mean Squared Error in every test. From an information theoretic point of view it is the natural loss for binary classification problems. When using Mean Squared Error none of the proposed weighting schemes improved over uniform weights whereas the opposite holds true for Binary Cross-Entropy where the best results are achieved using a combination of Median Frequency Balancing and Gradient Ratio.



Figure 1. Comparison of weighting methods. A sample image *(left)* and weight maps for Median Frequency Balancing *(left middle)*, Boundary Proximity *(right middle)* and Gradient Ratio *(right)*.

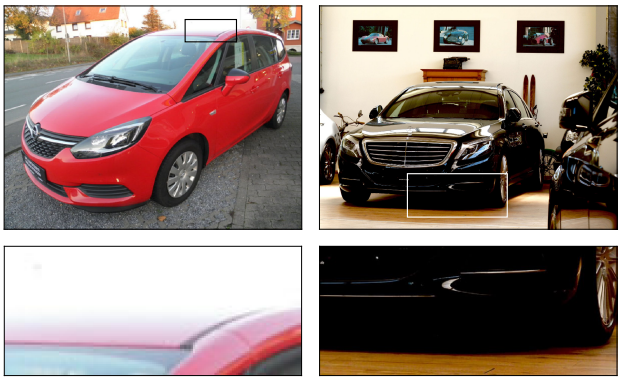


Figure 2. Examples of an overexposed bright region *(left)* and an underexposed dark region *(right)*.

	$\mathcal{L}_{\text{MSE}}$	$\mathcal{L}_{\text{BCE}}$
Default	$\mathbf{1.189 \times 10^{-2}}$	1.145×10^{-2}
MFB	1.284×10^{-2}	1.114×10^{-2}
Boundary Proximity	1.217×10^{-2}	1.126×10^{-2}
Gradient Ratio	1.296×10^{-2}	1.108×10^{-2}
MFB + Gradient Ratio	1.282×10^{-2}	$\mathbf{1.106 \times 10^{-2}}$

Table 1. Comparison of network performance after training with Mean Squared Error $\mathcal{L}_{\text{MSE}}$ and Binary Cross-Entropy $\mathcal{L}_{\text{BCE}}$ measured by $1-$Jaccard index.

4.2. Gradient Loss

We expand on the idea of using discrete gradients in loss functions and introduce the Gradient Loss. This loss is inspired by the H^1 Sobolev seminorm $|f|_{H^1} := \|\nabla f\|_{L^2}$. Instead of optimizing the plain value of the segmentation mask we optimize its gradient:

$$\mathcal{L}_{\nabla}(p, t) := \mathcal{L}_{\text{MSE}}(\nabla_\sigma p, \nabla_\sigma t) \tag{5}$$

In our tests we observe that neural networks trained with this loss produce masks with cleaner constant regions. We believe it allows them to learn that segmentation masks should be largely constant, i.e. for most areas $\nabla_\sigma p$ should be zero. It is not advisable to use the Gradient Loss on its own since it is based only on a seminorm. Depending on how the convolution treats missing values on the boundaries there might hold $\mathcal{L}_{\nabla}(p + c, t) = \mathcal{L}_{\nabla}(p, t)$ for constant values c (invariance to constant shifts). If the Gradient Loss is combined with Mean Squared Error we essentially obtain a discrete version of the H^1 Sobolev norm.

	$1-$Jaccard index
$\mathcal{L}_{\text{DSC}}$	9.237×10^{-3}
$\mathcal{L}_{\text{BCE}}$	1.145×10^{-2}
$\mathcal{L}_{\text{MSE}}$	1.189×10^{-2}
$\mathcal{L}_{\text{DSC}} + \mathcal{L}_{\text{BCE}}$	1.045×10^{-2}
$\mathcal{L}_{\text{DSC}} + \mathcal{L}_{\text{MSE}}$	9.633×10^{-3}
$\mathcal{L}_{\text{BCE}} + \mathcal{L}_{\nabla}$	1.082×10^{-2}
$\mathcal{L}_{\text{MSE}} + \mathcal{L}_{\nabla}$	1.224×10^{-2}
$\mathcal{L}_{\text{DSC}} + \mathcal{L}_{\text{BCE}} + \mathcal{L}_{\nabla}$	1.027×10^{-2}
$\mathcal{L}_{\text{DSC}} + \mathcal{L}_{\text{MSE}} + \mathcal{L}_{\nabla}$	$\mathbf{9.118 \times 10^{-3}}$
Previous solution	2.640×10^{-2}

Table 2. Network performance after training with composite loss functions measure by $1-$Jaccard index.

4.3. Composite Loss Results

It is common practice to combine the Dice Loss with a pixelwise loss [8] which results in segmentation maps with sharper boundaries. The combination of multiple loss functions is achieved by simple addition of the individual losses. Addition of the Dice Loss to the pixelwise losses uniformly results in a performance increase (see Table 2) due to its close relation to the Jaccard index. In these tests Mean Squared Error surpasses Binary Cross-Entropy by a significant margin while the incorporation of weighting schemes worsened results. The further addition of the Gradient Loss leads to mixed, but generally positive results. Although the effects on the combination of Dice Loss and Binary Cross-Entropy are minor, we achieve overall best results with the combination of the three losses Dice Loss, Mean Squared Error and Gradient Loss. The score outperforms the previous best result which was achieved with plain Dice Loss. Compared to a previous implementation used by the CarCutter service the segmentation error is reduced by 65%.

4.4. Postprocessing

The proposed weighting schemes and loss functions can only work if over- or underexposed regions are not completely devoid of texture. Otherwise a neural network may only learn to predict a pixel's probability to belong to the foreground class which inevitably causes non-sharp transition in the predicted masks. An alternate approach is the use of a custom postprocessing procedure.

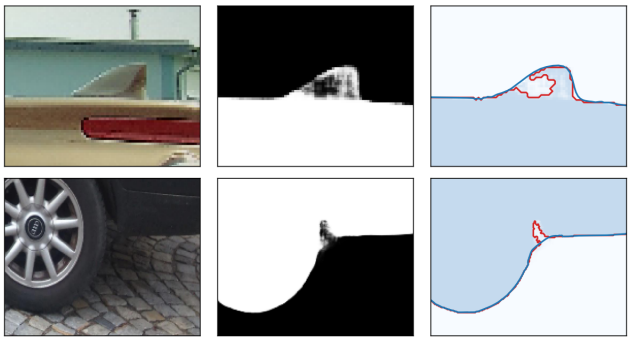


Figure 3. Two examples of active contour modeling. Crops of car images *(left)*, the inferred segmentation masks *(middle)* and contour segments *(right)* before active contour modeling *(red)* and after *(blue)*.

Inferred segmentation masks are thresholded and the resulting sharp separating contour between the foreground and background region is subjected to a refinement procedure. The contour is split into *certain* and *uncertain* regions depending on the neural networks certainty in its prediction. A contour region is considered to be certain if nearby values in the corresponding segmentation mask are close to 0 or 1, and uncertain otherwise.

Uncertain Regions These contour regions typically occur in over- or underexposed areas of an image and are iteratively adjusted using active contour modeling [7]. The method aims to minimize an energy functional of a spline contour in the inferred segmentation mask. Figure 3 shows the effects of the approach on two examples. The first example shows its positive influence while the second is a failure case.

Certain Regions Cars typically have large regions that are smooth for aerodynamic and aesthetic reasons. Edges that *are* present however can be rather sharp. The motivation of the following procedure, which we call *adaptive smoothing*, is to mimic this bias. We aim to perform a high degree of smoothing without displacing the contour by more than 0.5 pixels. The upper limit is enforce since based on the neural network assessment such a segment is already close to the ground truth target.

As an initial step the contour segment is split into separate sequences for x and y coordinates. The following procedure is applied separately to both. Let $\kappa = (\kappa_i)_{i=1}^N$ be such a sequence of real points. We use Gaussian filters G_{σ_i} with standard deviations σ_i that adept to the current position. A kernel G_{σ_i} is obtained by sampling a Gaussian density in the points $\mathbb{Z} \cap [-2\sigma_i, 2\sigma_i]$ and normalizing.

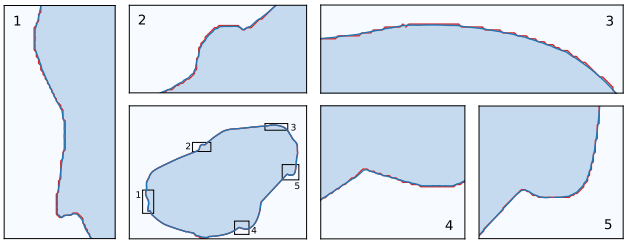


Figure 4. Comparison of a contour before postprocessing (red) and after adaptive smoothing (blue). Full segmentation mask and contours *(bottom second from the left)* and five enlarged regions.

The smoothed contour κ^s has equal shape to κ and is defined as

$$\kappa_i^s := \left(\kappa * G_{\sigma_i}\right)_i . \tag{6}$$

For the computation of the values σ_i we are looking for the largest kernel that displaces κ less than 0.5 pixels. A naive implementation of this idea has two issues: First the set $\left\{\sigma_i \in \mathbb{R}_{\geq 0} : \left|\kappa_i - \kappa_i^s\right| < 0.5\right\}$ might not be bounded and second this approach can lead to large jumps in consecutive entries of σ. For this reason we pose the definition with additional restrictions:

$$\begin{aligned} &(B) \quad \sigma_1 = \sigma_N = 0, \\ &(C) \quad |\sigma_i - \sigma_{i+1}| \leq \alpha, \quad i \in \{1 \dots N-1\}, \\ &(M) \quad \sigma_i \in \mathbb{R}_{\geq 0} \text{ maximal s.t. } |\kappa_i - \kappa_i^s| < 0.5. \end{aligned}$$

Under these conditions solutions exist and are unique. Requirement (B) enforces the fixed boundary conditions $\kappa_1 = \kappa_1^s$ and $\kappa_N = \kappa_N^s$ while (C) ensures continuity within the contour segment. The parameter α specifies an upper bound for the slope. In practice the setting $\alpha = 0.5$ performs well. For the implementation of this method it is advisable to only consider a discrete set of possible values for σ_i. A comparison of contours before and after postprocessing can be seen in Figure 4.

5. Conclusion

We studied methods for the generation of highly accurate binary segmentation masks, including weighting schemes that improved the performance of default loss functions and a novel Gradient Loss. In addition we developed a specialized postprocessing procedure that exploits a bias in our dataset. We created a solution that poses a significant upgrade over

a previous implementation of the CarCutter service. Direct comparison of masks generated by our implementation with ones generated by the service in April of 2019 showed a reduction of 65% in the segmentation error as measured by the Jaccard index. With the exception of our postprocessing procedure the presented methods are applicable to the general image segmentation task.

Acknowledgments

We would like to thank micardo GmbH[3] for a fruitful collaboration and the use of their private dataset. This work has been partly funded by the Austrian security research programme KIRAS of the Federal Ministry for Transport, Innovation and Technology (bmvit) under Grant 873495.

References

[1] V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *2017 IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2017.

[2] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The cityscapes dataset for semantic urban scene understanding. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3213–3223, June 2016.

[3] D. Eigen and R. Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV)*, ICCV '15, pages 2650–2658, Washington, DC, USA, 2015. IEEE Computer Society.

[4] Feng Ning, D. Delhomme, Y. LeCun, F. Piano, L. Bottou, and P. E. Barbano. Toward automatic phenotyping of developing embryos from videos. *2005 IEEE Transactions on Image Processing (TIP)*, 14(9):1360–1371, Sep. 2005.

[5] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[6] V. Iglovikov and A. Shvets. Ternausnet: U-net with VGG11 encoder pre-trained on imagenet for image segmentation. *CoRR*, abs/1801.05746, 2018.

[7] M. Kass, A. Witkin, and D. Terzopoulos. Snakes: Active contour models. *International Journal of Computer Vision*, 1(4):321–331, 1988.

[8] M. Khened, V. Alex Kollerathu, and G. Krishnamurthi. Fully convolutional multi-scale residual densenets for cardiac segmentation and automated cardiac diagnosis using ensemble of classifiers. *Medical Image Analysis*, 51, 01 2018.

[9] F. Milletari, N. Navab, and S. Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 565–571, Oct 2016.

[10] C. Rhemann, C. Rother, J. Wang, M. Gelautz, P. Kohli, and P. Rott. A perceptually motivated online benchmark for image matting. In *2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1826–1833, June 2009.

[11] O. Ronneberger, P.Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *2015 Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 9351 of *LNCS*, pages 234–241. Springer, 2015. (available on arXiv:1505.04597 [cs.CV]).

[12] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. Lecun. Overfeat: Integrated recognition, localization and detection using convolutional networks. In *2014 International Conference on Learning Representations (ICLR), CBLS, April 2014*, 2014.

[13] E. Shelhamer, J. Long, and T. Darrell. Fully convolutional networks for semantic segmentation. *2017 IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 39(4):640–651, Apr. 2017.

[14] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *2015 International Conference on Learning Representations (ICLR)*, 2015.

[15] J. Xu, H. Guo, A. Kageza, S. Wu, and S. AlQarni. Removing background with semantic segmentation based on ensemble learning. EAI, 9 2018.

[16] N. Xu, B. Price, S. Cohen, and T. Huang. Deep image matting. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 311–320, July 2017.

[3] https://www.micardo.com/ (accessed February 24, 2020)

Powder Bed Analysis in Additive Manufacturing Using Image Processing

Florian Recla, Martin Welk
UMIT – Private University for Health Sciences, Medical Informatics and Technology
florian.recla@edu.umit.at, martin.welk@umit.at

Abstract. *Systems for additive manufacturing are experiencing an enormous upswing in the industry. In this paper a method for the optical control of powder beds is presented. The system is based on a camera and directional lighting and is suitable for detecting two types of defects, including (i) areas where too little/too much powder has been applied, and (ii) areas with different porosity. The system is evaluated for both types of errors.*

1. Introduction

Binder-Jetting is a popular method for additive manufacturing of high-resolution components. In this process, powder is applied in layers, which is then selectively cured by a binder [3]. In order to prevent dead times and production downtimes in powder-bed-based additive manufacturing, a system was sought that would reliably find defects in the individual powder layers. Defects in the powder bed can occur either in the form of excess/missing powder, or as insufficient porosity of the powder. If such defects are not detected, components may be produced which do not achieve the expected strength values or contain predetermined breaking points inside. The analysis system should be simple in design and reliable in operation.

Three different types of optical analysis are used in existing plants: Laser triangulation [2], a camera with structured illumination [4] or a camera with directed illumination [1]. All approaches aim at creating a geometric image of the powder surface. Laser triangulation or structured illumination can be used to create three-dimensional models of the plane, while directed light can only be used to find qualitative deviations from the plane. Since a qualitative evaluation of the surface is sufficient, the method with directed light is chosen.

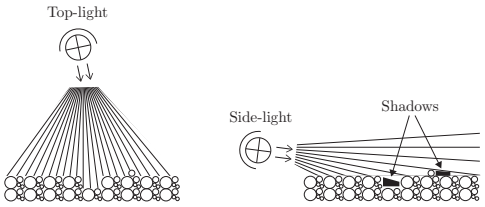


Figure 1. Side-light causes shadows at defective positions

2. Imaging System

The prototypical image system is installed in an existing machine for additive manufacturing.

The system consists of a camera and two lights. One light source illuminates the powder bed vertically from above to achieve the most uniform illumination possible (top-light). The second light source shines on the powder bed at a very flat angle (side-light). The side-light creates shadows when there are differences in height in the powder bed, which the camera captures from above. These shadows are not created when using the top-light (Figure 1). After the creation of each powder layer, two images are acquired, one using the top-light, and one using the side-light. By subtracting the two images from each other, the shadows are extracted and evaluated. The top-light ensures that color differences in the powder are not misinterpreted as shadows.

3. Image Processing

After acquiring the images of one layer, the processing of the images is done. The two images are high-pass filtered to minimise global illumination differences. (A constant correction is not possible because differences in the powder mixture lead to different reflection properties.) Afterwards the images are subtracted from each other to extract the shadows. A powder layer without defects thus produces an image with very low grey values, as there is no difference between top-light and side-light. Defects

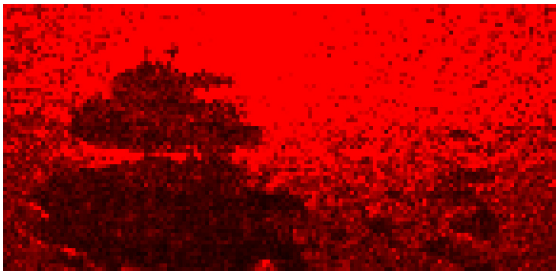


Figure 2. Variance of the difference image: high red intensity = high variance = missing powder

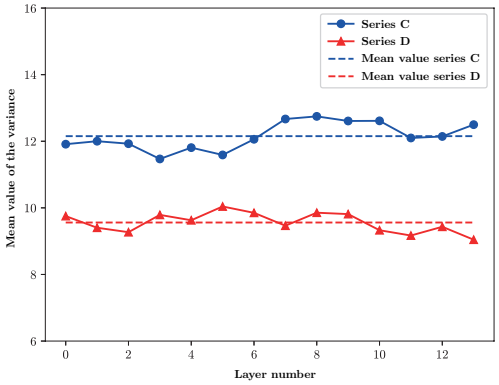


Figure 3. Variance of the grey values of the powder layers of two layers of test pieces with different porosities

or high porosity stand out from the image in the form of high grey values. The variance of the grey values in small, rectangular sectors is used as an indicator of quality. The observation in sectors is needed to locate the defects. A low variance indicates that few defects occur in the powder bed, the layer is well compacted. High values of the variance can indicate defects or high porosity.

3.1. Large defects

A difference image with powder missing over a large area can be seen in Figure 2. Areas with missing powder (or too much, albeit an unlikely case) produce large shadows, which appear as high grey values in the difference image. If the defects are larger than the sectors of the detection, the mean value of the grey values have to be considered in addition to the variance. A weighted average of the two factors is used as a criterion for the evaluation.

3.2. Porosity

Another factor that is evaluated is the porosity of the powder layers. The strength characteristics of the printed components are directly related to the degree of compaction of the powder. If the porosity is too high during the printing process, lower strength values are expected. Porous powder layers create shadows in the area of the grain size of the powder used, which appear as noise in the differential image. The variance of the grey values in the sectors is sufficient for detection. It should be noted that areas close to the light source tend to be overexposed, which makes evaluation more difficult.

4. Experimental Validation

To test whether the strength of components can be estimated during the manufacturing process, several series of test specimens were printed and then tested. It was confirmed that layers with few defects and a low porosity achieve higher strength values. The evaluation of the test is shown in Figure 3. Series C achieves $S_C = 10.8\,\mathrm{MPa}$ with a compression ratio of 10%, while series D achieves $S_C = 11.4\,\mathrm{MPa}$ with a compression ratio of 15%. Test specimens with higher density and better strength characteristics show lower porosity in the printing process, which is reflected in the differential images as a lower variance of the grey values.

5. Conclusion

Our tests have shown that a simple system consisting of one camera and two light sources is well suited for process control of powder bed based additive manufacturing processes. Both coarse defects in the powder bed and different porosities can be detected, which avoids production downtimes and enables quality control already while printing.

References

[1] T. Craeghs, S. Clijsters, E. Yasa, and J.-P. Kruth. Online quality control of selective laser melting. *22nd Annual International Solid Freeform Fabrication Symposium - An Additive Manufacturing Conference, SFF 2011*, 01 2011.

[2] M. Erler, A. Streek, C. Schulze, and H. Exner. Novel machine and measurement concept for micro machining by selective laser sintering. In *Proceedings of the International Solid Freeform Fabrication Symposium, Austin, TX, USA*, pages 4–6, 2014.

[3] I. Gibson, D. Rosen, and B. Stucker. *Additive Manufacturing Technologies – Rapid Prototyping to Direct Digital Manufacturing*, volume 5. 01 2010.

[4] Z. Li, X. Liu, S. Wen, P. He, K. Zhong, Q. Wei, Y. Shi, and S. Liu. In situ 3d monitoring of geometric signatures in the powder-bed-fusion additive manufacturing process via vision sensing methods. *Sensors*, 18(4):1180, 2018.

Grasping Point Prediction in Cluttered Environment using Automatically Labeled Data

Stefan Ainetter, Friedrich Fraundorfer
Graz University of Technology
{stefan.ainetter,fraundorfer}icg.tugraz.at

Abstract. *We propose a method to automatically generate high quality ground truth annotations for grasping point prediction and show the usefulness of these annotations by training a deep neural network to predict grasping candidates for objects in a cluttered environment. First, we acquire sequences of RGBD images of a real world picking scenario and leverage the sequential depth information to extract labels for grasping point prediction. Afterwards, we train a deep neural network to predict grasping points, establishing a fully automatic pipeline from acquiring data to a trained network without the need of human annotators. We show in our experiments that our network trained with automatically generated labels delivers high quality results for predicting grasping candidates, on par with a trained network which uses human annotated data. This work lowers the cost/complexity of creating specific datasets for grasping and makes it easy to expand the existing dataset without additional effort.*

1. Introduction

Automated grasping is a very active field of research in robotics. The process of having a robot manipulator successfully grasp objects in a cluttered environment is still a challenging problem. Recent state-of-the-art for grasping position computation often use deep learning techniques and supervised learning. However, these methods usually need to be trained on a large amount of labeled data. Therefore, it is of high interest to find techniques to automatically label data for robotic grasping. Previous work [17, 19] focused on using raw RGBD data for automatic object segmentation by leveraging sequential depth information from the scene. However, the segmentation mask is not sufficient as annotation for grasping point prediction because many state-of-the art approaches define the grasping proposal using a bounding box representation.

We propose a fully automatic pipeline from raw RGBD data to a system that predicts grasping point candidates using our automatically labeled data for training. Figure 1 shows our workflow. As practical example, we captured RGBD data from log ordering in the wood industry. We will demonstrate the usefulness of our approach by training a deep neural network to predict grasping points using our automatically generated labels as ground truth. The main contributions of this work are:

1. A fully automatic annotation pipeline for grasping point prediction using sequential RGBD data.

2. An automatic annotation method that allows dense labeling of grasping points for graspable objects. Additionally, the annotations contain implicit information about the order of object removal due to the usage of sequential input data. These labels can be directly used for training a supervised learning approach.

3. A deep neural network which is able to predict grasping points in a cluttered environment, solely trained with a small number of automatically labeled images.

2. Related Work

Grasping point detection. The conventional method for grasping point detection uses information about object geometry, physics models and force analytics [1]. With the rise of deep learning, data-driven methods [2] became more common. Methods like [13, 9, 7, 20] use deep neural networks and supervised learning to predict multiple grasping points for a single object. Chu et al. [4] were able

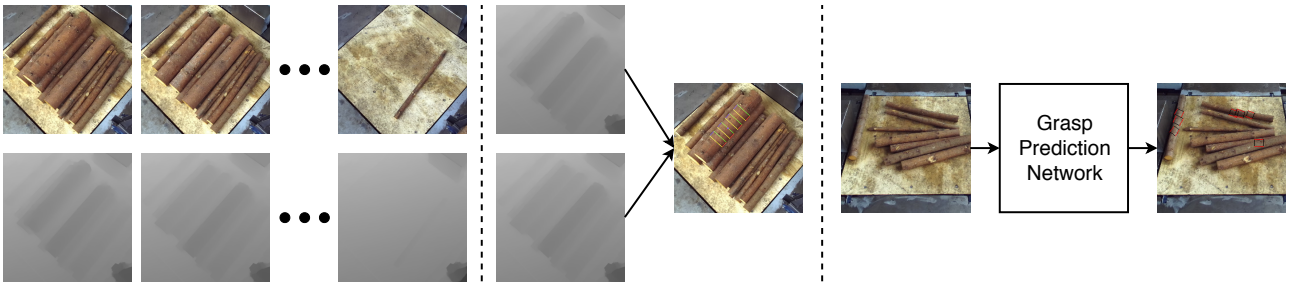


Figure 1. Overall workflow of our method containing data acquisition, automatic grasping point annotation using depth images and training a deep network for grasping point prediction. (**Left**) Our dataset is constructed by recording sequences of RGBD images while a human expert removes wooden logs from the scene. (**Middle**) The sequence of captured depth images is used to automatically annotate grasping points in every corresponding RGB image. (**Right**) This automatically annotated data are then used to train a deep neural network to predict grasping points.

to predict multiple grasping points for multiple objects in an image. Zeng et al. [18] showed that they are able to grasp unseen objects with their winning contribution for the Amazon Robotics Challenge in 2017. Other approaches [12, 10] use Reinforcement Learning (RL) on a real or simulated robot to perform thousands of grasp attempts and use the feedback to improve the grasping point predictions. RL has the advantage that no labeled data are necessary for training, but it is on the other hand very time and hardware consuming.

Representations of grasping points in 2D. Saxena et al. [16] described a grasping point as $g = \{x, y\}$, where x and y define the center of the grasping point proposal. This representation lacks information about the opening width of the gripper. Redmon and Angelova [13] overcame this limitation by using a rectangular representation for the grasping point. This is very similar to the bounding box representation of objects in the field of object detection, with the addition of a rotation angle θ which describes the orientation of the bounding box. An overview about other common representations can be found in [3].

Automatic label generation. Datasets used for deep learning are often hand annotated, which is time consuming and can be error prone due to the involvement of human annotators. In the domain of object segmentation, modern tools like DeepExtremeCut [11] or GrapCut [15] significantly reduce the amount of work for labeling RGB data to a small number of clicks. However, they are not fully automatic and are not able to work with depth data. Zeng et al. [19] showed that they are able to use background subtraction to generate segmentation masks of new objects in the scene. Suchi et al. [17], most similar to our approach, use sequences of depth images to predict segmentation masks of the objects in the scene. However, the difference of our method compared to all previously mentioned approaches is that we do not only calculate the segmentation mask, but directly infer grasping proposals. Furthermore, segmentation masks do not give any information in which order the objects should be removed, which can be crucial for grasp success in cluttered environment.

3. Data Acquisition and Automatic Annotation

This section describes our simple strategy to automatically label grasping points for scenes with objects in a cluttered environment.

3.1. Data Acquisition Protocol

The process requires a statically mounted RGBD camera which records color and depth information from the scene. We then ask human experts to remove one object after the other from the scene. After each successful grasp, we capture depth and color images. Figure 2 shows a sequence of recorded RGB images. This method provides us not only with consecutive RGBD images of the picking procedure, but also gives implicit information about the optimal order of object removal according to a human expert. This information is highly important because not all objects are equally easy to grasp due to their random placement (e.g. objects on top of one another).

3.2. Automatic Label Generation

As illustrated in Figure 3, we perform automatic grasping point annotation through an 3-stage pipeline. Our algorithm takes two consecutive depth images from the scene as input and calculates grasp proposals for the object which was removed. A grasp



Figure 2. Sequence of recorded RGB images. The sequence starts in the top left with the full stack of objects and we record an RGB image after each object removal. We also record the corresponding depth image for every RGB frame.

proposal $\mathbf{g}$ is defined as

$$\mathbf{g} = \{x, y, \theta, w, h\}, \tag{1}$$

where x and y describe the center of the grasp proposal, θ describes the angle of the rotated bounding box, and w and h describe the width and height of the predicted box.

Initial depth segmentation. The main focus of our algorithm is to detect depth changes in the scene after a successful grasp was performed by a human expert. Therefore, we calculate the depth difference I^* of two consecutive depth images as

$$I^* = |I_1 - I_2|, \tag{2}$$

where I_1 and I_2 are the depth images previously normalized between 0 to 255. The output I^* is a rough estimate of the segmentation mask of the removed object.

Segmentation mask refinement. The intermediate segmentation is coarse and contains noise mainly due to inaccurate sensor values and small movements of the objects. Therefore, further refinement of the segmentation mask is needed. We apply binary image morphology to remove the majority of noise and smooth the mask edges. A Gaussian filter is then applied for further noise reduction and to create the refined mask which is used for further processing. The Gaussian filter g_{filter} is defined as

$$g_{filter}(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}, \tag{3}$$

where x and y are the spatial dimensions of the intermediate mask I^*, and σ is defined as the standard deviation for the Gaussian kernel. In our experiments, we set $\sigma = 1$, which means that it is equal for both axes.

Automatic grasping point annotation. The refined segmentation mask is then used to calculate geometric features of the object. The skeleton of the object mask is calculated by using [8] to remove boarder pixels as long as the connectivity does not break. The resulting skeleton of the object is approximated with a line segment, which makes it more robust to outliers. Each point on this line segment can then be used as a possible center of a grasp proposal. The height h and the rotation angle θ of a grasp proposal is determined by calculating the intersection between a line, which is normal to the skeleton and passes through the center of a grasp proposal, and the edges of the mask. The bounding box width w is directly dependent on the used gripper and we set this parameter manually to suit our robotic gripper. All this information are then combined and used to generate the final grasping proposals. The proposals have certain characteristics:

1. The center of a bounding box is located at the spine of the object.
2. The height of the bounding boxes are bounded to the edges of the object mask.
3. The width of the bounding boxes can be set manually, because this parameter highly depends on the gripper characteristics.
4. The majority of the grasp proposals are generated near the center of mass, which is based on the assumption that these points more likely lead to an successful grasp.

Results. Our automatic annotation pipeline allows us to generate a high number of grasping labels without any supervision of human annotators. Furthermore, due to the fact that the data is recorded while an expert did the grasping, we implicitly have supervision about which object should be removed from

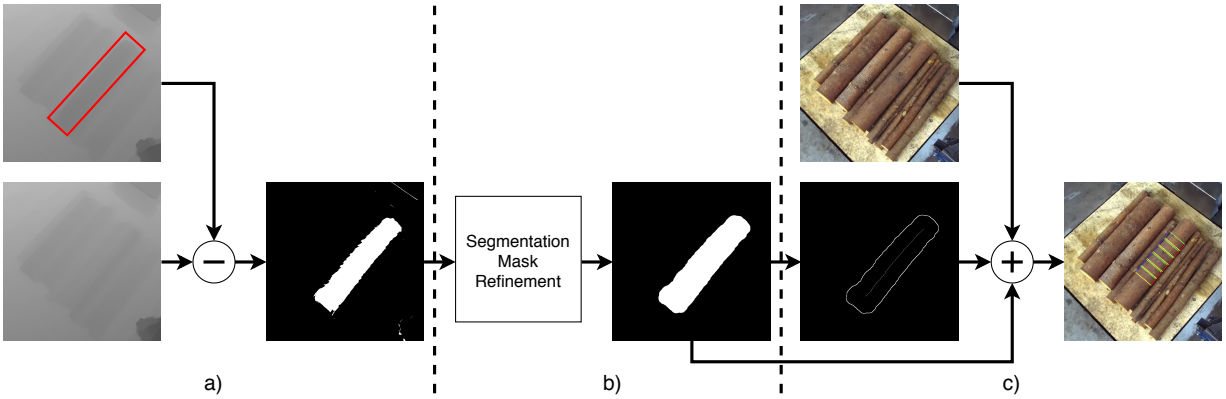


Figure 3. Our automatic annotation pipeline. a) Two consecutive depth images with one object removed (marked in red). Calculating the difference of the depth images gives a rough segmentation mask of the removed object. b) Refinement of the mask using morphological operations and Gaussian filtering. c) Geometric features (object edges, skeleton, center of mass) are calculated using the refined segmentation mask and are used afterwards to calculate the final position of the grasping point proposals. The last step transfers the proposed bounding boxes to the corresponding RGB image.

the scene, without any additional costs. The only time humans are involved is, when checking all the predicted labels via manual inspection to find images which contain erroneous labels. In this process we roughly drop 10% of the images to avoid inaccurate labeled training data. Figure 4 shows results of our automatically labeled dataset.

3.3. Human-based Data Annotation

Additionally to our automatic labeling approach, we also labeled the whole dataset manually. The idea is to train a grasp prediction network on both types of labels independently, and then compare the performance of both approaches. All hand labeled data were checked by human experts with domain knowledge to verify the correctness of the annotations.

4. Grasping Point Prediction in a Cluttered Environment

Chu et al. [4] proposed a deep neural network to predict multiple grasping points for multiple objects in the scene. We adapted their approach and retrained the network with our specific dataset.

4.1. Network Architecture and Loss Function

The network architecture is based on the Faster R-CNN object detection framework [14] using a ResNet-50 [6] as backbone. It takes a three channel RGB image as input and predicts a number of grasping point candidates, whereas one candidate $\mathbf{g}$ is defined as described in Equation 1. Note that the rotation angle θ is quantized into $R = 19$ intervals, which makes the prediction of this parameter a classification problem. All other parameters (see Equation 1) are predicted using regression. During training, the composite loss function $\mathcal{L}_{total}$ is defined as

$$\mathcal{L}_{total} = \mathcal{L}_{gpn} + \mathcal{L}_{gcr}, \tag{4}$$

where $\mathcal{L}_{gpn}$ describes the loss according to the grasp proposal net and $\mathcal{L}_{gcr}$ is the grasp configuration prediction loss. The loss term $\mathcal{L}_{gpn}$ is used to define initial rectangular bounding box proposals without orientation ($\{x, y, w, h\}$), whereas $\mathcal{L}_{gcr}$ is used to define the orientation and the refined bounding box prediction $\{x, y, \theta, w, h\}$. Figure 5 shows the structure of the prediction network and indicates how the loss parts $\mathcal{L}_{gpn}$ and $\mathcal{L}_{gcr}$ are calculated. Further information about the network architecture and the loss function can be found in [4].

4.2. Data Preprocessing and Augmentation

Our dataset for training the prediction network consists of only 52 images. Therefore, data augmentation is used to increase the size of the training data by the factor of 100. Figure 6 shows examples of the augmented data. This increases the variation in the training data and decreases the possibility of overfitting during training. After augmentation, each image was resized to $227 \times 227px$ to fit the input dimension of the network.

4.3. Training Schedule

Pre-trained ImageNet [5] weights are used as initialization for the ResNet-50 backbone to avoid overfitting and ease the training process. All other layers beyond ResNet-50 are trained from scratch. The whole structure of the network can be seen in Figure 5. We used the Adam Optimizer and trained our



Figure 4. Visualization of automatically generated labels. Each edge of one grasping point proposal is visualized with a different color to show the orientation of the box. Our method allows dense labeling of the object but only four grasping point proposals are visualized in each image to guarantee the clarity of the visualization. Note that only one object per image is labeled which implicitly adds expert knowledge about the optimal order of object removal.

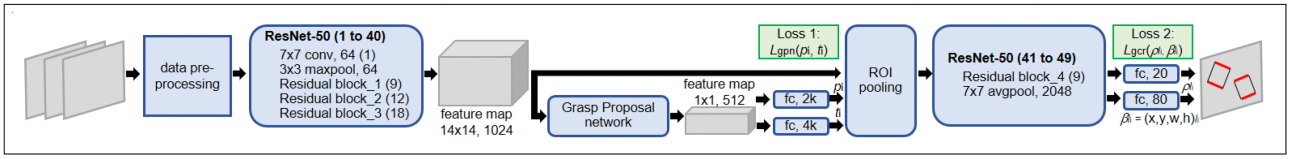


Figure 5. Architecture of the grasping point prediction network. The network takes RGB images as input, and predicts multiple grasping candidates. The grasping candidates are defined as an oriented rectangular bounding box. The output bounding boxes are drawn with different colors, whereas the red edges denote the parallel plates of the gripper and the black lines indicate the opening width of the gripper. Figure was taken from [4].



Figure 6. Data Augmentation. (**Left**) RGB input image, (**others**) randomly shifted and rotated input image.

network for 50000 iterations with a initial learning rate $\alpha = 0.0001$. The anchor sizes for the bounding box proposals are chosen according to the size of the objects in our dataset using $[8, 16, 24, 28]px$, with anchor ratios of $[0.5, 1, 2]$. All other hyperparameters were taken from [4]. Note that the goal of these experiments was to show the practical benefit of our method for automatic label generation, rather than to compete for the best possible performance for grasping point prediction. We believe that a more careful selection of hyperparameters, combined with an optimized training schedule could further boost the results.

5. Experiments and Evaluation

We trained the previously described prediction network two times separately, once with automatically annotated data and once with the same data labeled by hand. Both networks were evaluated using a test set containing 22 images which are independent from the training data (different camera position, random placement of objects) to verify the generalization capabilities of our network. We used the same training schedule for both methods, as well as the same parameters for non-maximum suppression for both experiments to ensure a fair comparison. The evaluation of our predicted grasping candidates is divided into two parts:

1. Quantitative evaluation of the predicted grasping points by calculating the ratio of graspable / non-graspable candidates.

2. Qualitative evaluation by visualizing the predicted grasping candidates.

5.1. Quantitative Evaluation

For quantitative evaluation we decided to calculate the relative number of predicted grasp candidates that are non-graspable for both networks trained with manually/automatically labeled data. We define a non-graspable prediction as 1) the size of the predicted bounding box is unsuitable (either too big or too small) or 2) grasping is not feasible due to partial occlusion of the object. Figure 8 shows examples of non-graspable candidates. Table 1 shows the quantitative results indicating that a deep network trained with automatically labeled data can achieve similar performance compared to the same network trained with manually labeled data.

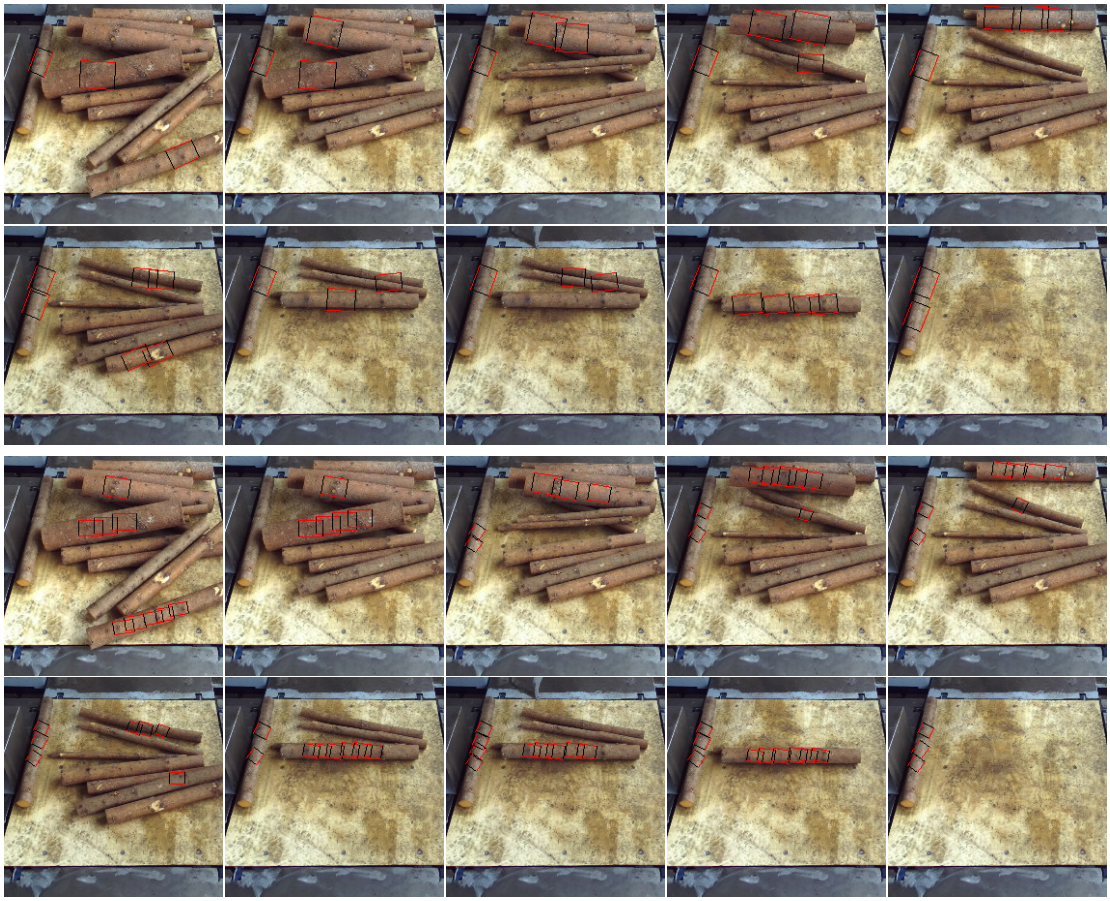


Figure 7. Comparison of predicted grasping candidates for both networks trained on automatically labeled data (**top two rows**) and manually labeled data (**bottom two rows**). We apply non-maximum suppression to reduce the number of visualized boxes and to ensure the clarity of the visualization.

Method	Valid grasping candidates in %
Auto-Label	81.17
Man-Label	83.43

Table 1. Relative number of valid grasping candidates for both approaches. The network trained with automatically labeled data is named **Auto-Label**, whereas the network trained with manually labeled data is named **Man-Label**. Both networks show similar performance which emphasizes the usefulness of our automatically labeled data.



Figure 8. Examples for non-graspable predictions. (**Left**) predicted bounding box not graspable because another object is on top; (**middle**) box too big; (**right**) box too small.

5.2. Qualitative Results

Qualitative results of our grasping point predictions are shown in Figure 7 for the networks trained with the manually annotated data and the automatically generated labels respectively.

6. Conclusion

We have proposed an automatic annotation method for easily generating grasp proposals for robotic manipulations using only one RGBD camera. Our annotation method requires minimal human interaction and is highly cost effective. With the proposed method, we generated ground truth data and successfully trained a deep neural network to predict grasping candidates. To underline the usefulness of our approach, we trained our grasp prediction network with hand annotated and automatically annotated data separately, and our experiments showed similar performance for both attempts. This leads to the conclusion that our automatically generated labels are highly accurate.

We believe that the best strategy to train a deep network for grasping point predictions is to initially train with a large number of automatically annotated frames using our method, and afterwards fine-tune it with a small number of frames annotated by human experts. This strategy can lead to highly accurate results with minimal human interaction.

References

[1] A. Bicchi and V. Kumar. Robotic grasping and contact: A review. In *2000 International Conference on Robotics and Automation (ICRA)*, volume 1, pages 348–353. IEEE, 2000.

[2] J. Bohg, A. Morales, T. Asfour, and D. Kragic. Data-driven grasp synthesis—a survey. *IEEE Transactions on Robotics*, 30(2):289–309, 2013.

[3] S. Caldera, A. Rassau, and D. Chai. Review of deep learning methods in robotic grasp detection. *Multimodal Technologies and Interaction*, 2(3):57, 2018.

[4] F.-J. Chu, R. Xu, and P. A. Vela. Real-world multiobject, multigrasp detection. *IEEE Robotics and Automation Letters*, 3(4):3355–3362, 2018.

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009.

[6] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[7] S. Kumra and C. Kanan. Robotic grasp detection using deep convolutional neural networks. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 769–776, 2017.

[8] T.-C. Lee, R. L. Kashyap, and C.-N. Chu. Building skeleton models via 3-d medial surface axis thinning algorithms. *CVGIP: Graphical Models and Image Processing*, 56(6):462–478, 1994.

[9] I. Lenz, H. Lee, and A. Saxena. Deep learning for detecting robotic grasps. *The International Journal of Robotics Research*, 34(4-5):705–724, 2015.

[10] S. Levine, P. Pastor, A. Krizhevsky, J. Ibarz, and D. Quillen. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *The International Journal of Robotics Research*, 37(4-5):421–436, 2018.

[11] K.-K. Maninis, S. Caelles, J. Pont-Tuset, and L. Van Gool. Deep extreme cut: From extreme points to object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 616–625, 2018.

[12] L. Pinto and A. Gupta. Supersizing self-supervision: Learning to grasp from 50k tries and 700 robot hours. In *2016 International Conference on Robotics and Automation (ICRA)*, pages 3406–3413. IEEE, 2016.

[13] J. Redmon and A. Angelova. Real-time grasp detection using convolutional neural networks. In *2015 International Conference on Robotics and Automation (ICRA)*, pages 1316–1322. IEEE, 2015.

[14] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[15] C. Rother, V. Kolmogorov, and A. Blake. " grabcut" interactive foreground extraction using iterated graph cuts. *ACM transactions on graphics (TOG)*, 23(3):309–314, 2004.

[16] A. Saxena, J. Driemeyer, and A. Y. Ng. Robotic grasping of novel objects using vision. *The International Journal of Robotics Research*, 27(2):157–173, 2008.

[17] M. Suchi, T. Patten, D. Fischinger, and M. Vincze. Easylabel: A semi-automatic pixel-wise object annotation tool for creating robotic rgb-d datasets. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 6678–6684. IEEE, 2019.

[18] A. Zeng, S. Song, K.-T. Yu, E. Donlon, F. R. Hogan, M. Bauza, D. Ma, O. Taylor, M. Liu, E. Romo, et al. Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching. In *2018 International Conference on Robotics and Automation (ICRA)*, pages 1–8. IEEE, 2018.

[19] A. Zeng, K.-T. Yu, S. Song, D. Suo, E. Walker, A. Rodriguez, and J. Xiao. Multi-view self-supervised deep learning for 6d pose estimation in the amazon picking challenge. In *2017 International Conference on Robotics and Automation (ICRA)*, pages 1386–1383. IEEE, 2017.

[20] X. Zhou, X. Lan, H. Zhang, Z. Tian, Y. Zhang, and N. Zheng. Fully convolutional grasp detection network with oriented anchor box. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7223–7230, 2018.

The Difficulties of Detecting Deformable Objects Using Deep Neural Networks

Nikola Djukic, Markus Vincze
Automation and Control Institute, TU Wien, Vienna, Austria
{dukic,vincze}@acin.tuwien.ac.at

Walter G. Kropatsch
Pattern Recognition and Image Processing Group, TU Wien, Vienna, Austria
krw@prip.tuwien.ac.at

Abstract. *Object detectors based on deep neural networks have revolutionized the way we look for objects in an image, outperforming traditional image processing techniques. These detectors are often trained on huge datasets of labelled images and are used to detect objects of different classes. We explore how they perform at detecting custom objects and show how shape and deformability of an object affect the detection performance. We propose an automated method for synthesizing the training images and target the real-time scenario using YOLOv3 as the baseline for object detection. We show that rigid objects have a high chance of being detected with an AP (average precision) of 87.38%. Slightly deformable objects like scissors and headphones show a drop in detection performance with precision averaging at 49.54%. Highly deformable objects like a chain or earphones show an even further drop in AP to 26.58%.*

1. Introduction

Object detection in RGB images has received a lot of attention in the previous years due to advances in deep neural networks (DNN) research. Classical techniques usually rely on searching for features in an image that were hand-crafted by a human. Deep neural networks on the other hand use huge datasets of hand-labelled images to learn these features. These labels are either a bounding box of an object or its mask. This approach has shown great efficiency. In general there are two types of DNN based object detectors. The first group performs the detection in a single run through a network. These methods are generally fast and can even run in real-time



Figure 1. Objects used for evaluation

with standard hardware. Second group has a separate region proposal and detection stage, which usually makes the execution of the methods slower but more precise than the first group of methods. Recently, a combination of CBNet and Cascade R-CNN has achieved a new state of the art result on the COCO dataset [9] with an impressive AP50 of 71.9%. [10]

Detecting custom objects is a common problem in robotics. DNN or more precisely Convolutional Neural Networks (CNN) require large amounts of data for training. Having that data hand-labelled by a human is extremely time consuming so there is a lot of research going on in the field of synthesizing training data. This is typically done by first making a 3D reconstruction of the objects and then placing them in a virtual environment which allows the simulation of artificial deformations and the creation of arbitrary synthetic views where labels are taken from the 3D template. However, obtaining a full 3D reconstruction is not possible with all objects, especially

in the case of deformable objects. Objects like folding headphones, scissors, chains, cables can vary in appearance depending on their current usage. This poses a problem for CNN based object detectors. We propose a simple RGB based method for recognition of rigid but also deformable objects and synthesize images for training a neural network. We then test this method by training the YOLOv3 [13] network with the fully synthetic dataset and explore how does the shape of an object, ie. it's symmetry and deformability affect the detection performance.

The contributions of the work include:

- An automated pipeline on synthetic data generation used for detection and recognition of both rigid and deformable objects.
- A novel RGB based method for quick and effortless acquisition of object masks.
- We explore the effect of deformability of an object to its detection performance.

2. Related work

Computer vision tasks depend on large amounts of annotated training data. For the tasks of detecting object classes such as cars or airplanes there are numerous hand-annotated datasets available: COCO [9], PASCAL VOC [3] and Open Image Dataset [7]. These datasets are built by researchers or companies and consist of a large number of images. Each image has annotations of objects of interest. This may be a bounding box only or contain the mask of the object as well. The COCO (Common Objects in Context) dataset consists of over 330 thousand images containing objects that are split into 80 classes. However, sometimes, especially in robotics related tasks, we are interested in detecting a specific object. For example not any mug but the user's favourite coffee mug. The mentioned datasets are of little use in these cases, so there is a necessity for a specialized dataset. Datasets are normally difficult to obtain so there is a lot of research concerning synthesizing datasets.

Jungwoo Huh et al. [6] proposed a method for synthesizing training data that, similarly to ours, relies on obtaining masks of an object. In order to produce the synthetic images they use pure pasting, whereas we use a combination of pasting and Poisson image editing. Additionally they evaluate their method on rigid objects only, for example a baseball bat, a bottle, a toy rifle etc. The only deformable object that they use is an umbrella but they keep it closed during the training and testing so we can consider it as a rigid object in this case. Additionally, they use YOLOv2, which has a lower mAP (Mean Average Precision) than the YOLOv3 while also preserving the ability to process the images in real-time. For obtaining the masks of the objects they use a semi-automatic segmentation method while ours is fully automated and does not involve any manual post-processing.

Debidatta Dwibedi et al. [2] assume that object images, which cover diverse viewpoints, are available. They apply a CNN to obtain a mask of the object. They then randomly place the object into a scene image using Poisson cloning. Next, they train the Faster R-CNN [14] network using the synthetic images and evaluate the method on the GMU-Kitchens dataset [5]. For the evaluation of the method they also use exclusively rigid objects like bottles, detergents, cups, cornflakes packages etc. Although simple, the method achieves an mAP of 88%, which is similar to what we report on detection of rigid objects.

Georgakis et al. [4] propose a method for synthesizing training data that takes into consideration the geometry and semantic information of the scene. They use publicly available RGB-D datasets, the GMU-Kitchens [5] and the Washington Washington RGB-D Scenes v2 [8], as backgrounds for the object images. Using RANSAC they detect planes in the image and artificially place objects on top of them, while also scaling their size according to the distance from the camera. This method produces natural looking images, because instead of being placed randomly in an image, the objects such as a cup or a bottle are placed on a flat desk surface or on the ground. They test their method using SSD and Faster R-CNN [14] and report an mAP between 70% and 85% depending on how much real data they use. Considering the fact that the scenes they use for evaluation are cluttered this is a good results. The objects used for evaluation are a bowl, a cup, a cereal box, a coffee mug and a soda can. These are all non deformable objects.

3. Synthetic Data Generation

Object detection is required in cases such as self-driving cars, unmanned aerial vehicles, robotics etc. Except for detecting rigid objects like cars, chairs or cups it is often needed to detect deformable ob-

jects like chains or cables. Most of the previous work on object detection focuses on detecting rigid objects[3, 15, 6, 2]. Our goal is to expand this research to deformable objects as well. We train an object detector based on CNN to detect both rigid and deformable objects. For this task a big amount of training images is required. Obtaining this data manually is time consuming, therefore we propose a method for synthesizing the training data which includes an RGB based segmentation procedure that is able to handle deformable objects. We then use publicly available datasets as background for the synthetic images and augmentation techniques to increase the variability of the dataset.

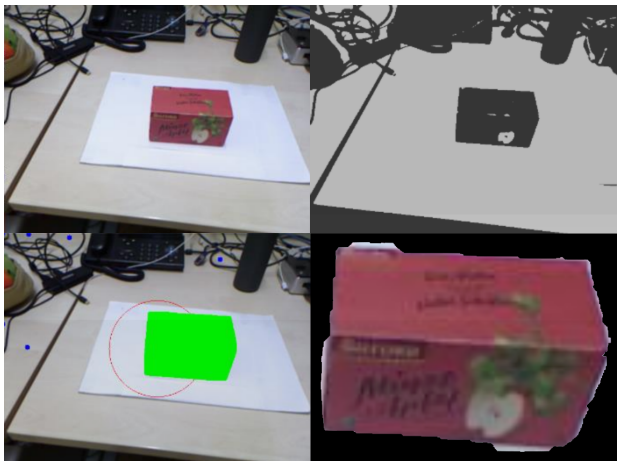


Figure 2. Illustration of the mask acquiring process. Top left image shows the original RGB image. Top right image shows the result of appyling k-means method to the original RGB image. Bottom left image shows the automatically selected contour and the area inside of it colored in green. Bottom right image shows the final extracted object masks.

3.1. Data acquisition

Publicly available datasets which contain annotated objects are suitable for training CNN to detect object classes. However, when it comes to detecting specific objects, a specialized dataset is required. We synthesize a dataset by capturing the images of the objects and develop a method to segment them from the flat surface on top of which they were placed.

For the recording of objects a Kinect camera by Microsoft mounted on a tripod is used. The camera is placed at approximately 30 cm above the flat surface and facing the object at an angle of approximately 45 degrees. During the recording, both the camera and the flat surface are stationary. The flat surface should preferably be unicolor so that the object is clearly distinguishable from it.

After the recording was initiated, the object was manipulated by hand in order to get it to face the camera from all possible viewing angles. The point is to get the object to face the camera in as many unique perspectives as possible. The advantage of this method is that it is able to capture deformable objects by simply changing their shape while they are being recorded.

3.2. Data processing

In order to synthesize images that are needed for training of the network object masks are needed. Obtaining the masks of the object is possible by manually segmenting the object from the background or by using a segmentation method. Manually segmenting objects is inefficient, therefore we devise a simple method for object segmentation that is used for both rigid and deformable objects. For the segmentation of the object from the background a combination of computer-vision based methods is used. It contains the following five steps:

1. Firstly, k-means clustering is applied to the image with the k value of 2. This method is successful at distinguishing the boundaries of interest. Additionally it is computationally more efficient than a possible alternative of using Otsu's Thresholding.

2. After application of k-means, morphological operations like image closing and erosion are applied to the image in order to connect possible discontinuities in the border of the object.

3. Next, contour detection is applied to the whole image and locations of gravity centers of the area inside of the detected contours are determined. A red circle is drawn on the image coming from the Kinect camera, which is shown on the screen, in which the center of the object should be placed in order to automatically start the capturing process.

4. The algorithm then determines if the contour satisfies conditions in terms of its length and distance from the center of the image and, if that is the case, the recording is started. After the capturing process is initiated a predetermined number of object projections is recorded at a regular time interval or per keyboard command. The number of projections recorded is

40, which is usually more than enough to capture the object from all different angles.

5. If the object of interest is deformable the capturing process is paused, to capture a deformed state, and then re-started. Images of segmented objects are then stored on a hard drive for synthesizing training data.

Figure 2 shows the illustration of the mask acquiring process.

3.3. Synthesizing training data

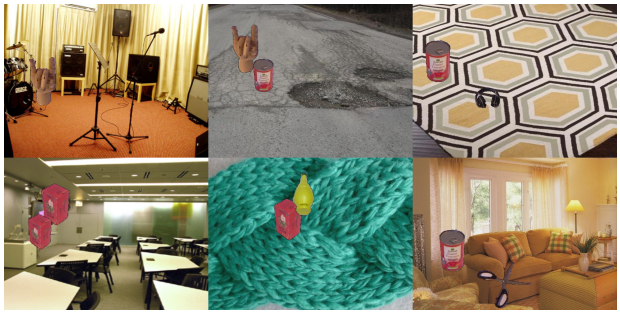


Figure 3. Examples of synthetic images that are used for training the YOLOv3 network

In order to synthesize the training images we used a combination of Poisson image cloning [11] and pure pasting of the segmented objects onto different background images. As background for the synthetic images we used the Indoor Scene Recognition dataset [12] and Describable Textures Dataset (DTD) [1]. We used ten different objects for the evaluation and generated 2500 synthetic images per object. To handle the blur that appears while the objects are moving, we artificially blurred 20% of the images by adding horizontal motion blur between 5 and 15 pixels to the objects. As objects move closer or further away from the camera their relative size changes, so we introduce artificial scaling of the object uniformly distributed between 50% and 125% of its original size. In order to tackle the occlusion problem small patches of textures from the DTD dataset are placed randomly on 10% of the synthetic images. These cover between 0% and 50% of the object surface. Additionally we introduce multiple objects to the image and allow them to occlude each other by a maximum IOU (Intersection Over Union) of 40%.

Figure 3 shows the examples of the synthetic images that are used for training the YOLOv3 network.

4. Evaluation

To evaluate the method, we trained the CNN based object detector YOLOv3 using the synthetic images. A total of ten objects were used, which differ greatly in their shape and deformability. We know already that YOLO performs very well when facing rigid objects. Therefore our aim was to explore to what extend the shape of an object can be deformed. As an example of rigid objects we use a can, two different tea boxes, and a lemon juice bottle. Slightly deformable are headphones, scissors and a human hand model. Extremely deformable objects that we used are earphones, power cable and a piece of chain.

Properties of objects used for evaluation and their detection precision are presented in Table 1.

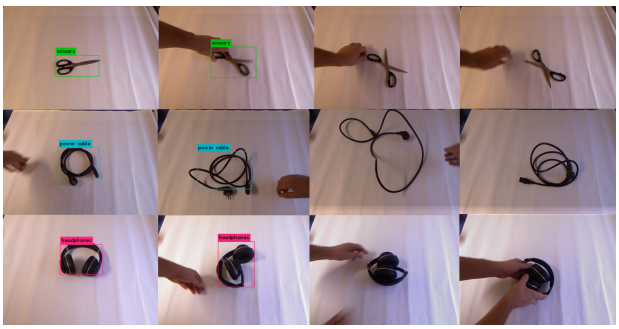


Figure 4. Successful and unsuccessful cases of detection of different deformable objects

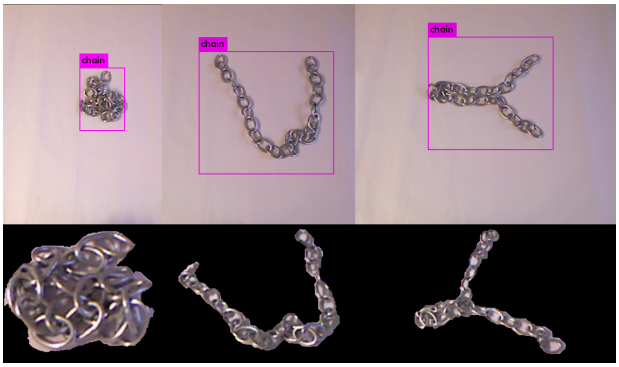


Figure 5. Chain detection success cases



Figure 6. Chain detection failure cases

In order to evaluate the precision of the proposed method two minute videos of each object being manipulated were filmed and every 20th frame extracted

Objects	Deformability	Precision
lemon juice bottle	rigid	89.61
red tomato can	rigid	84.52
red tea box	rigid	87.65
yellow tea box	rigid	87.76
headphones	slightly def.	57.32
scissors	slightly def	54.15
human hand model	slightly def.	37.16
power cable	highly def.	34.66
chain	highly def.	30.24
earphones	highly def.	14.86

Table 1. Object detection performance, def - deformable

and manually annotated. We then ran the YOLO network trained with the synthetic data and calculated precision for each object, taking as ground truth the hand-annotated data. An Intersection Over Union (IOU) of 50% was considered a successful detection.

As shown in previous work rigid objects like a can, a tea box or lemon juice bottle have a very good chance at getting detected with the precision being at close to 90%. These objects do not change greatly in appearance when placed in different positions and it is therefore easy for the network to learn their appearance. We purposely choose that some of the objects have similar color, so that, due to lack a of great number of objects used for evaluation, the detection performance may not be attributed to simple color searching.

Slightly deformable objects that we used were scissors, headphones and a human hand model. We see that in the case of slightly deformable objects the detection performance drops significantly with it being around 55% for the scissors and the headphones. The chances of detecting the human hand model are even lower, being 37.16%.

The last three objects that we evaluate are a chain, a power cable and a pair of earphones. These objects are considered to be highly deformable. Again there is a clear drop in detection performance with the precision of earphones detection being only 14.86%. Chances that a power cable or a piece of a chain will be detected are a bit over 30%.

All of the objects used for evaluation can be seen in Figure 1. Detection of objects used for evaluation using YOLO trained on COCO dataset was unsuccessful for all of the objects except for the scissors with the detection rate of 62.35%, similar to our result. The chain detection success cases can be seen on Figure 5, whereas the chain failure cases are presented in Figure 6.

We can see that the in the cases where chain detection is successful a mask of chain taking a similar structure can be found in the bottom row of Figure 5. In the cases where the chain detection fails there are no masks available that resemble the given chain structure.

We then pose the chain detection problem as single link detection problem and try to detect the structure of the chain by detecting each individual link in the chain. In order to do so, we use our proposed method to segment the link in many different orientations and synthesize the training images. We then connect the individual links into a chain and test detection of individual links while the chain is taking different configurations. The results of a single link detection can be seen on the top row of the Figure 7.

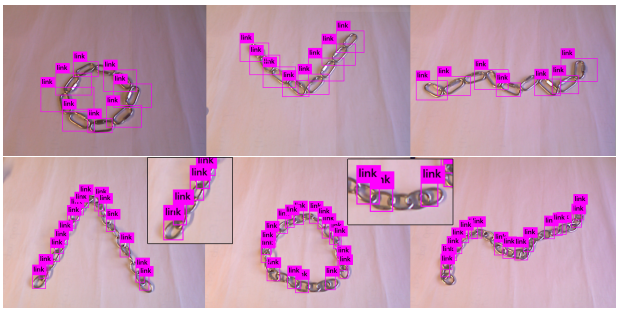


Figure 7. Examples of link detection

We also record 100 images of a chain taking different shapes and manually annotate each of the links on the chain in all of the images and train the YOLOv3 network with those annotated images. The results of a single link detection with the manually annotated links can be seen on the bottom row of the Figure 7.

We took chain as an example of a highly deformable object that is made out of simple rigid elements. These results show that the detection of an deformable object is possible by detecting its elementary parts.

Successful and unsuccessful cases of object detection are presented in the Figure 4. As shown, on the examples of the power cable, the scissors and the headphones, detection is successful in some of the configurations. If the configuration is slightly changed the detection fails. This is due to the big variability in the appearance of these objects which is caused by their deformability. Potentially, modelling of deformable objects such as a power cable or a chain could be used to generate big amounts of dif-

ferent object masks. This would enable the network to learn a bigger amount of object views, than those that a human demonstrator can show in a reasonable time.

Our method works well when facing rigid objects, when the number of unique views is limited. However, when it comes to deformable objects, number of unique views increases dramatically. Therefore, in those cases the efficiency of our method drops significantly.

5. Conclusion

In this paper we intend to highlight open problems of a standard object detector when applied to slightly and highly deformable objects. We specifically trained the YOLOv3 detector to cope with these cases. To reduce the time consuming effort of image annotations, we proposed an automated method for synthesizing the training images. The idea is to show objects on simple background and use a short videos and a few annotations with augmentation of training data to obtain better performance. While this works well for rigid objects with an AP of 87.38%, we show that for slightly deformable objects like scissors and headphones the detection performance drops significantly to 49.54%. The drop is, as expected even more drastic for highly deformable objects like a chain or earphones, down to AP of 26.58%.

Using the example of a chain we show that it is possible to pose the problem of detection of the deformable objects as detection of its elementary rigid element - a link. To further tackle this problem, modelling of deformable objects could be used for synthetic data generation.

Acknowledgment

This research is partially supported by the Vienna Science and Technology Fund (WWTF), project RALLI (ICT15-045 and Festo AG & Co. KG.

References

[1] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[2] D. Dwibedi, I. Misra, and M. Hebert. Cut, paste and learn: Surprisingly easy synthesis for instance detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1301–1310, 2017.

[3] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010.

[4] G. Georgakis, A. Mousavian, A. C. Berg, and J. Kosecka. Synthesizing training data for object detection in indoor scenes. *arXiv preprint arXiv:1702.07836*, 2017.

[5] G. Georgakis, M. A. Reza, A. Mousavian, P.-H. Le, and J. Košecká. Multiview rgb-d dataset for object instance detection. In *2016 Fourth International Conference on 3D Vision (3DV)*, pages 426–434. IEEE, 2016.

[6] J. Huh, K. Lee, I. Lee, and S. Lee. A simple method on generating synthetic data for training real-time object detection networks. In *2018 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 1518–1522, Nov 2018.

[7] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, T. Duerig, and V. Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *arXiv:1811.00982*, 2018.

[8] K. Lai, L. Bo, and D. Fox. Unsupervised feature learning for 3d scene labeling. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3050–3057. IEEE, 2014.

[9] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[10] Y. Liu, Y. Wang, S. Wang, T. Liang, Q. Zhao, Z. Tang, and H. Ling. Cbnet: A novel composite backbone network architecture for object detection. *arXiv preprint arXiv:1909.03625*, 2019.

[11] P. Pérez, M. Gangnet, and A. Blake. Poisson image editing. *ACM Transactions on graphics (TOG)*, 22(3):313–318, 2003.

[12] A. Quattoni and A. Torralba. Recognizing indoor scenes. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 413–420, June 2009.

[13] J. Redmon and A. Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[14] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[15] K. Wang, F. Shi, W. Wang, Y. Nan, and S. Lian. Synthetic data generation and adaption for object detection in smart vending machines. *arXiv preprint arXiv:1904.12294*, 2019.

Border Propagation: A Novel Approach To Determine Slope Region Decompositions

Florian Bogner
TU Wien
e1225415@student.tuwien.ac.at
Co-First Author

Alexander Palmrich
TU Wien
apalmrich@gmail.com
Co-First Author

Walter G. Kropatsch
TU Wien
krw@prip.tuwien.ac.at
Supervisory Author

Abstract. *Slope regions are a useful tool in pattern recognition. We review theory about slope regions and prove a theorem linking monotonic paths and the connectedness of levelsets. Unexpected behavior of slope regions in higher dimensions is illustrated by two examples. We introduce the* border propagation *(BP) algorithm, which decomposes a d-dimensional array ($d \in \mathbb{N}$) of scalar values into slope regions. It is novel as it allows more than 2-dimensional data.*

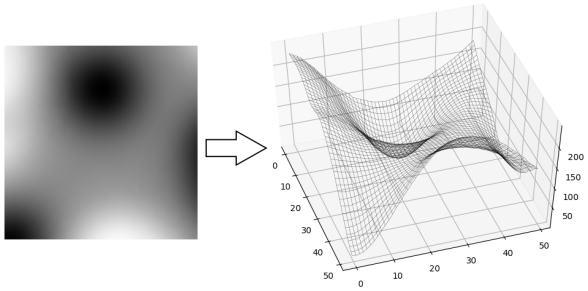


Figure 1. gray-scale to height-map conversion

1. Introduction

In this section we develop an intuitive understanding of the term *slope region* [3] and its generalization to higher dimensions. The concise definition of the terms already employed here is reserved for the next section.

Consider an image, either gray-scale or in color. If it is a color image, it can be decomposed into its color channels (red-green-blue), which can individually be read as gray-scale images. We consider pixel intensity of one such gray-scale image as the height of a landscape, yielding a 2D surface in 3D space. The surface will have peaks in areas where the image is bright, and will have dales in dark areas.

Now our aim is to partition the surface into *regions* (i.e. subsets) in a particular way: We require each region to consist only of a single slope, by which we mean that we can ascend (or descend) from any given point of the region, to any other given point of the region, along a path that runs entirely within the region. Such a decomposition is not unique, but we can at least try to get a partition *as coarse as possible*, meaning that we merge slope regions if the resulting subset is still a slope region, and we iterate this until no further change occurs. There might be many different coarsest slope decompositions.

The criterion we used to describe slopes, any two points being connected by either an ascending or a descending path, can easily be used in higher dimensions. Think of a computed tomography scan, which will yield gray-scale data, but not just on a 2D image, but rather on a 3D volume. We want to partition the 3D volume, such that any two points in a region can be connected via an either ascending or descending path within the region. Recall that *ascending* and *descending* refers to the intensity value of the tomography scan as we move in the volume. For piecewise linear functions on a volume, decompositions were introduced in [1].

By abstracting from image and tomography to a real function $f : \Omega \to \mathbb{R}$ defined on some subset of $\mathbb{R}^n$ (think of it as the pixel intensity function), and by rigorously defining a coarsest slope decomposition, we can lift the concept to arbitrary dimensions in a mathematically concise fashion.

2. Defining Slope Regions

In this and the following chapters we will consider a topological space $(\Omega, \mathcal{T})$ with a continuous function $f : \Omega \to \mathbb{R}$. In practice or for ease of imagination, $(\Omega, \mathcal{T})$ will typically be a rectangle or cuboid subset of $\mathbb{R}^2$ or $\mathbb{R}^3$ equipped with the eu-

clidean topology and f will describe a continuous image or 3D-scan.

Definition 2.1. A *path* is a continuous function from the real interval $[a, b]$ (with $a < b$) into a topological space Ω.

Definition 2.2. Two points $x \neq y$ in a topological space Ω are called *path-connected* if and only if there exists a path $\gamma : [a, b] \to \Omega$ with $\gamma(a) = x$ and $\gamma(b) = y$.

Definition 2.3. The set of all points which are path-connected to a point $x \in \Omega$ is the *connected component of* x:

$$[x] := \{y \in \Omega | x \text{ is path-connected to } y\}$$

Any subset of Ω which can be written in above way (for a suitable choice of x) is called a *connected component*.

Definition 2.4. A path $\gamma : [a, b] \to \Omega$ is called *monotonic* if and only if the whole path is ascending or the whole path is descending, meaning the first or second formula below has to hold, respectively:

$$\forall s, t \in [a, b] : s < t \Rightarrow f(\gamma(s)) \leq f(\gamma(t))$$
$$\forall s, t \in [a, b] : s < t \Rightarrow f(\gamma(s)) \geq f(\gamma(t))$$

Definition 2.5. Let $R \subset \Omega$. R is called *slope region* or *monotonically connected* if and only if for all $x, y \in R$ there exists a monotonic path $\gamma : [a, b] \to R$ with $\gamma(a) = x$ and $\gamma(b) = y$.

Definition 2.6. A family of sets $\{A_i \subset \Omega \mid i \in I\}$ is called a *slope region decomposition* if and only if:

- A_i is a slope region for all $i \in I$
- $\forall i, j \in I : i \neq j \Rightarrow A_i \cap A_j = \emptyset$.
- $\bigcup_{i \in I} A_i = \Omega$

Definition 2.7. Consider two slope region decompositions $\mathcal{A} = \{A_i \subset \Omega \mid i \in I\}$ and $\mathcal{B} = \{B_j \subset \Omega \mid j \in J\}$. We call $\mathcal{A}$ *coarser than* $\mathcal{B}$, alternatively $\mathcal{B}$ *finer than* $\mathcal{A}$, in Symbols $\mathcal{A} \succeq \mathcal{B}$ if and only if

$$\forall j \in J \, \exists i \in I : B_j \subset A_i.$$

Theorem 2.8. *$\succeq$ is a partial order, i.e. fulfills reflexivity, antisymmetry and transitivity.*

Proof: Straight forward. Antisymmetry follows from the decomposition property. □

Definition 2.9. A slope region decomposition $\mathcal{A}$ is called *maximally coarse* or simply *coarse* if and only if there is no other coarser slope region decomposition.

We can apply Zorn's lemma [6] to the partial order $\succeq$, which yields the existence of maximal elements. For this we need to show that chains have upper bounds.

Theorem 2.10. *For any ascending chain of slope region decompositions $(\mathcal{A}_i)_{i \in I}$, that is $t \geq s \Rightarrow \mathcal{A}_t \succeq \mathcal{A}_s$, there is a slope region decomposition $\mathcal{A}_\infty$ satisfying $\forall i \in I : \mathcal{A}_\infty \succeq \mathcal{A}_i$.*

Proof: We consider the equivalence relation "*connected in $\mathcal{A}_i$*" for two points $x, y \in \Omega$:

$$x \sim_i y :\Leftrightarrow \exists A \in \mathcal{A}_i : x \in A \wedge y \in A$$

The equivalence relation is a subset of Ω^2, and $\mathcal{A}_t \succeq \mathcal{A}_s$ implies $\sim_t \supset \sim_s$. This suggests the use of $\sim_\infty := \bigcup_{i \in I} \sim_i$ to get an upper bound. Indeed the equivalence classes of $\sim_\infty$ yield a partition $\mathcal{A}_\infty$ of Ω, which is coarser than any $\mathcal{A}_i$. But do they form a slope region decomposition? Yes: For any two fixed points x, y to be $\sim_\infty$-connected, they need to be $\sim_i$-connected for some $i \in I$. So there is a monotonic path linking x and y in $A = [x]_{\sim_i} \subset [x]_{\sim_\infty}$, by which they are monotonically connected in $\mathcal{A}_\infty$. Therefor $\mathcal{A}_\infty$ is a slope region decomposition. □

Hence every set Ω has a coarse decomposition.

Theorem 2.11. *Let $A \subset \Omega$ be a path-connected set. A is a slope region if and only if all levelsets of f in A are path-connected, i.e.*

$$\forall c \in \mathbb{R} : f^{-1}(\{c\}) \cap A \text{ is path-connected.}$$

Proof: "⇒" via contraposition:

Suppose there exists a $c \in \mathbb{R}$ with $L := f^{-1}(c) \cap A$ not path-connected. We decompose L in its components and pick x and y from different components. Since $f(x) = f(y) = c$ a monotonic path between x and y would have to lie completely in L. However, since x and y are from different components, they cannot be connected by a path in L and therefore cannot be connected with a monotonic path. Therefore, A is not a slope region.

"⇐" via *ironing out* an arbitrary path:

Given $x, y \in A$ we have to find a monotonic path γ. Without loss of generality suppose $f(x) \geq f(y)$. Since A is path-connected, there exists an (not necessarily monotonic) path $\gamma_0 : [a, b] \to A$ from x to

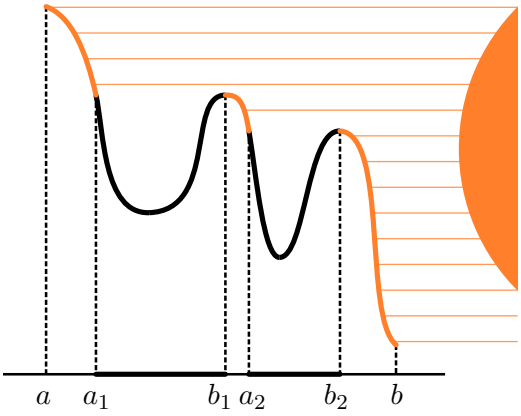


Figure 2. applying the Rising Sun lemma

y. Using the Rising Sun lemma [5] on the continuous function $f \circ \gamma_0$ we get the *shadow* $S = \bigcup_{i \in I}(a_i, b_i)$ consisting of at most countably many intervals.

S consists of the points which contradict the monotonicity of $f \circ \gamma_0$, thus we want to *iron out* these points.

Let $c_n := f(a_n) = f(b_n)$. Since the levelset of c_n is path-connected, we can connect $\gamma_0(a_n)$ and $\gamma_0(b_n)$ with a level path $\gamma_n^* : [a_n, b_n] \to A$.

Finally, we define:

$$\gamma(\sigma) := \begin{cases} \gamma_n^*(\sigma) & \sigma \in [a_n, b_n] \\ \gamma_0(\sigma) & \text{elsewhere} \end{cases}$$

γ is a monotonic path from x to y, thus A is a slope region. □

3. Results In The Plane And Counterexamples In Higher Dimensions

There are two theorems ([3] Lemma 1 and [3] Lemma 2) that are useful, but only hold if $\Omega \subset \mathbb{R}^2$, not in general if $\Omega \subset \mathbb{R}^d$ for $d > 2$. But first, we prove a lemma.

Theorem 3.1. *Let A be a slope region. Then the closure $\bar{A}$ is also a slope region.*

Proof: Follows from continuity of f. □

The following theorem is only formulated for closed slope regions, but because of the above theorem this is not a big restriction.

Theorem 3.2. *Let $d = 2$ and $A \subset \mathbb{R}^d$ be a closed and bounded slope region. Let $(\partial A_i)_{i \in I}$ be an enumeration of the connected components of the boundary ∂A. For $i \in I$, if ∂A_i is homeomorphic to a circle, then $f|_{\partial A_i}$ has at most one local minimum and one local maximum (but the extrema might be spread out in a connected plateau).*

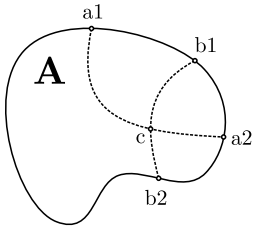


Figure 3. a sketch of the situation in Theorem 3.2

Proof: Assume there are two local minima $a_1, a_2 \in \partial A_i$ with $f(a_1) \leq f(a_2)$. Since ∂A_i is homeomorphic to a circle, there have to be local maxima b_1 and $b_2 \in \partial A_i$ between them with $f(a_2) < f(b_1) \leq f(b_2)$, one on each arc.

Since A is a slope region, a_1 and a_2, as well as b_1 and b_2 have to be connected by a monotonic path. Because of the Jordan Curve Theorem [2, p.169], these paths have to cross in a point $c \in A$. But this yields a contradiction: $f(c) \leq f(a_2) < f(b_1) \leq f(c)$. Thus the assumption of the existence of two local minima has to be false. □

Note: The circle assumption is actually unneccesary and the proof without it remains the same in spirit, but becomes inhibitivly technical, which is why we omit it here.

Example 3.3. Let $\Omega = \mathbb{R}^3$ and $A = B_1(0,0,0)$ be the closed unit ball. Let f be the distance to the x-Axis.

$$f : \mathbb{R}^3 \to \mathbb{R} : (x, y, z) \mapsto \sqrt{y^2 + z^2}$$

The levelsets of f in A are either the x-Axis for $f \equiv 0$ or the sides of cylinders for $f > 0$. In any case, they are connected. Thus, by Theorem 2.11, A is a slope region. ∂A has one connected component, which is the unit sphere. $f|_{\partial A}$ has two local minima, which are the intersections with the x-Axis, $(1,0,0)$ and $(-1,0,0)$.

Thus, the previous theorem does not hold in $\mathbb{R}^3$. In fact, it does not hold in any $\mathbb{R}^d$ for $d > 2$. There is also no limit on the number of local minima on the surface of a slope region.

Theorem 3.4. *Let $d = 2$ and $A \subset \mathbb{R}^d$ be a slope region. Let $s \in A$ be a saddle point. Then, $s \in \partial A$.*

Proof: Assume s is an interior point of A, which means there is a open set U with $s \in U \subset A$. s being a saddle point means there is a neighborhood $V \subset U$ so that $V_- := V \cap [f < f(s)]$ as well as $V_+ := V \cap [f > f(s)]$ decompose into two or more connected components.

Pick a_1, a_2 from different components of V_- as well as b_1, b_2 from different components of V_+. a_1 and a_2 have to be connected by a monotonic path, but this path has to move outside of V since the points are from different components of V_- and by virtue of being monotonic, the path cannot go through V_+. Analogue for b_1 and b_2.

Again by the Jordan Curve Theorem, these two paths have to cross in some point c, which again yields a contradiction.

$$f(c) \leq max(f(a_1), f(a_2)) < \\ < f(s) < min(f(b_1), f(b_2)) \leq f(c)$$

Thus the assumption that s is a interior point has to be false. □

Example 3.5. Let $\Omega = A = \mathbb{R}^3$. Let f be the distance from the unit circle laying in the x-y-plane.

$$f : \mathbb{R}^3 \to \mathbb{R} : (x, y, z) \mapsto \\ \begin{cases} \left\| \left(x - \frac{x}{\|x,y\|_2}, y - \frac{y}{\|x,y\|_2}, z \right) \right\|_2 & \|(x, y)\|_2 \neq 0 \\ \|(1, 0, z)\|_2 & \|(x, y)\|_2 = 0 \end{cases}$$

Again, let us look at the levelsets to show A is a slope region. The levelset of $f \equiv 0$ is the unit circle. For $0 < f < 1$ the levelsets are tori. $f \equiv 1$ marks a transition and the levelset is a torus with its hole closed. Then, for $f > 1$ the levelsets look like the exterior surface of a self intersecting torus, topologically equivalent to a sphere. All these levelsets are connected. Thus, A is indeed a slope region.

Now consider the point $(0, 0, 0)$. Along the x and y-direction it is a local maximum, however along the z-direction it is a local minimum. Thus, it is a saddle point. Therefore, theorem 3.4 does not hold in higher dimensions.

4. Motivating The Border Propagation (BP) Algorithm

Now we will work our way to the central insights on which the border propagation algorithm (*BP*) hinges. Let us develop ideas for smooth (hyper-)surfaces first, and deal with discrete variants in the next section.

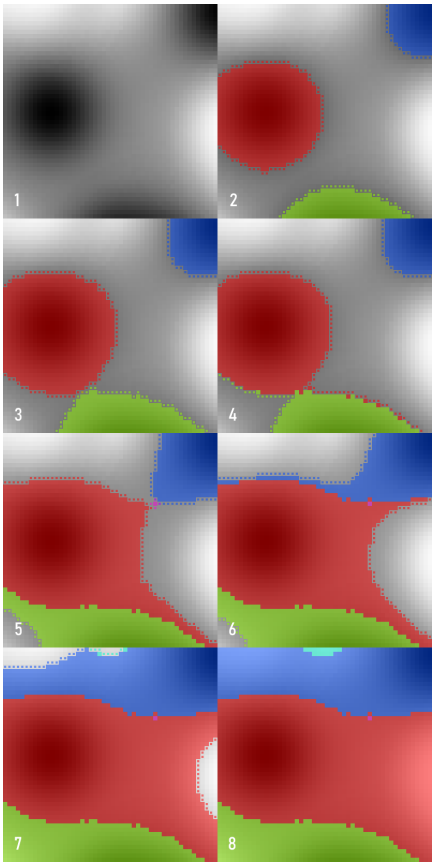


Figure 4. evolution of discretized regions during the algorithm

Slope regions can be constructed and grown in a straight-forward iterative manner by sweeping through the function values from lowest to highest. This is similar to the intuition employed in Morse theory[4, Section 1.4]. Visualize a smooth, compact 2D surface in 3D space. We want to decompose this surface into slope regions. Initially, our decomposition is empty, i.e. there are no slope regions (thus we don't have an actual *decomposition* yet). This is shown in Figure 4, Image 1.

Imagine a water level rising from below the surface, up to the point of first contact. Starting at this global minimum, we add a new region, containing only the argmin (i.e. a single point on the 2D image where the minimal value is taken).

Now, there might be many points where the global minimum is taken. This will either be due to a connected region (*plateau*) on the surface, which we want to include into the single existing region, or it will be due to individual dales, which all have their lowest point at the same height. In this case, we can't

put the points into the existing region, because we would not be able to get from one argmin to another via a monotonic path. Instead, we need to add a new region for each individual dale.

Both cases can be dealt with by contracting connected points into their connected component, and creating a new region for each resulting component. This will ensure that plateaus are assigned to a single region.

As the water rises, we can add points to an existing region growing it upwards if they are just outside the region[1]. Otherwise they correspond to a distant local minimum and have to be dealt with as before, by opening a new region for each point (or rather, each connected component). This is shown in Figure 4, Image 2.

With the water rising further still, the regions will grow upwards to a point where they meet (Figure 4, Image 3). Any such point is a saddle point, and we have to account for it the next time we want to grow any one of the touching regions. The saddle point connects the edges of the regions which meet in it, at the current height of the water level. It might be the case that the not-yet-assigned points (the ones above the water) get separated into multiple connected components, or they might remain connected.

If the points remain connected, then we decide for a single one of the involved regions to be allowed to grow upwards from the component. This means that one region effectively inherits the growth directions from the other region(s). The other region(s) lose their potential for expansion and remain frozen in their current state.

If the unassigned points have multiple components (as in Figure 4, Image 3: the unassigned grey points are separated into the lower left and upper right areas), then we may assign one component to each involved region. The regions will then grow only in the directions determined by the assigned components as the water rises. This can be observed in Figure 4, Image 4: The green region is allowed to grow to the lower left, while the red region floods the upper right. The same procedure of swapping areas of expansion also happens as we move from Image 5 to 6. Any region without an assigned component remains frozen. Should there be more components than regions, then we open up a new region for each surplus component, as in the top of Image 7.

An oddity which can occur are self-loops: A region might grow into a ”C”-shape, and then proceed to close up into an ”O”-shape. This case can be treated similarly as above, the only difference is that the saddle is found by recognizing that the region collides with itself, not with another region.

Eventually this procedure will arrive at the global maximum, and the entire surface will be divided into regions. Since we proceeded with the necessary care and attention along the way, we ensured that the regions remained slope regions, and we also only created additional regions when we absolutely had to, showing that the resulting composition is maximally coarse.

The same algorithm can be applied in higher dimensions. We deal with iso-hyper-surfaces as level sets, but the topological considerations about connectedness remain the same as in the illustrative 2D case.

5. Discrete BP

The somewhat vague description of BP in the previous section assumed a continuous surface. In most applications, however, the data will be provided in a discrete raster format. Some intricacies arise from this discretization, most notably iso-surfaces of a smooth function f will not have a straight-forward representation in the discrete grid obtained from rasterizing f. The data structure we use is a set of indices, representing the positions in the discrete array already assigned to a region.

Each region also has a set of (yet) unassigned points, which determine where the region might grow in the next iteration, called the *border*. This effectively models the smooth levelsets in the discrete representation.

The pseudo-code for the algorithm is printed below, the executable python code can be accessed in our github repository: *https://github.com/SirFloIII/MustererkennungLVA*

6. Further Potential Development

The result of BP is satisfactory, but we assume that improvements can be made in running time. The code was profiled multiple times and has been adapted to run faster with significant gains in many instances.

[1] Why can we do that? By adding only points which are connected to the region we ensure path-connectedness, and by growing the region upwards, we can construct ascending paths from old points to new ones. The smoothness of the surface guarantees that while moving at a fixed height, we can reach all points of the region with that height.

Algorithm 1 Border Propagation

Enumerate all values of f and collect points into levelsets.
for each levelset in bottom to top order **do**
  Add points to regions if they are in the border of a region
  if an added point is in the border of different region **then**
    Find the union of the borders of the involved regions
    Find the connected components thereof
    Assign these to the regions in an arbitrary way
  end if
  if an added point splits the border of the region in two **then**
    Reduce the border the region to one component
    for each other component **do**
      Create a new region containing the component as border
    end for
  end if
  for leftover points that cannot be added to any regions **do**
    Create a new region containing only that point
  end for
end for

Additional features we consider:

- Providing a *tolerance* parameter, which governs how steep a continuous function might get, before an iso-surface is deemed disconnected in the discrete data. This would allow for a trade-off between continuous connectedness and discrete connectedness. Modeling continuous connectedness creates fewer slope regions and yields pleasing results on smooth data, but the resulting regions are not monotonically connected (in the discrete sense of *connected*) in general. Discrete connectedness guarantees monotonic connectedness, but it necessarily creates significantly more and smaller slope regions. On smooth data the latter tends to produce too fine of a decomposition.
- Using established data structures that model smooth level sets from discrete data. There might be performance gains in employing such a data structure.

7. Conclusion

In this paper we have shown that slope regions of continuous functions in high dimensions ($n \geq 3$) do not have the same critical point properties well-established in 2D. Hence previous graph-based methods of building slope region decompositions by merging regions according to their border extrema will fail in high dimensions. Instead we developed a new, levelset-based method of growing regions, which yields slope region decompositions on discrete data of arbitrary dimension.

Acknowledgements

The BP algorithm as well as this paper are the result of a pattern recognition course held at the *Vienna University Of Technology* from Oct 2019 till Jan 2020. Professor W. KROPATSCH introduced us to the concept of slope regions and posed the challenge to compute them in high dimensions (> 2). We wish to thank him as well as DARSHAN BATAVIA and the anonymous reviewers for their valuable input.

References

[1] H. Edelsbrunner, J. Harer, V. Natarajan, and V. Pascucci. Morse-smale complexes for piecewise linear 3-manifolds. In *Proceedings of the nineteenth annual symposium on Computational geometry*, pages 361–370, 2003.

[2] A. Hatcher. *Algebraic Topology*. Cambridge University Press, 2002.

[3] W. G. Kropatsch, R. M. Casablanca, D. Batavia, and R. Gonzalez-Diaz. Computing and reducing slope complexes. In *International Workshop on Computational Topology in Image Context*, pages 12–25. Springer, 2019.

[4] Y. Matsumoto. *An introduction to Morse theory*. Iwanami series in modern mathematics. American Math. Soc., Providence, RI, 2002.

[5] F. Riesz. Sur un Théoreme de Maximum de Mm. Hardy et Littlewood. *Journal of the London Mathematical Society*, 1(1):10–13, 1932.

[6] M. Zorn. A remark on method in transfinite algebra. *Bulletin of the American Mathematical Society*, 41(10):667–670, 1935.

How High is the Tide? Estimation of Flood Level from Social Media

Julia Strebl, Djordje Slijepcevic, Armin Kirchknopf,
Muntaha Sakeena, Markus Seidl, Matthias Zeppelzauer
St. Pölten University of Applied Sciences, Austria
{firstname.lastname}@fhstp.ac.at

Abstract. *The availability of social media data represents an opportunity to automatically detect and assess disasters to better guide emergency forces. We propose a method for flood level estimation from user-generated images to support assessing the severity of flooding events. Furthermore, we provide labeled data for water detection. Results on a public benchmark dataset are promising and motivate further research.*

1. Introduction

The visual estimation of flood levels is a novel task. In this paper we aim at detecting images with a certain water level, i.e. where the water is at least knee-high. Our work is based on preliminary work from the MediaEval 2019 Satellite Task [1]. Our contribution is twofold: we demonstrate the feasibility of visual flood level estimation by combining a supervised water detector with pose estimation and we provide novel image annotations for water detection.

Related work focuses on either visual, textual or multimodal flood level estimation from social media content [1]. Zaffaroni et al. [5], for example, combine multiple pre-trained networks for the estimation of flood level. Further approaches can be found in [4]. We aim at presenting a simple and efficient approach to provide a baseline for future comparison.

2. Methods

Input to our approach are social media images. We propose two approaches that build upon three components: (i) a supervised water detector that predicts whether a certain image or image region contains water, (ii) a pose estimator that detects people and their joints and (iii) a rule-based fusion module that combines the information from the water detector and the pose estimator to make a final decision. The first approach (see Figure 1A) aims at detecting water within the whole image and detecting at least one person with concealed lower body parts. The second approach (see Figure 1B) performs water detection locally around each detected human body. If at least for one body the model detects concealed lower extremities and water in the vicinity, the image is assigned to knee-high water.

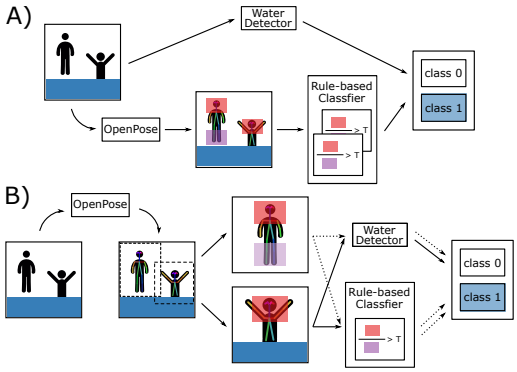


Figure 1: Global (A) and local (B) approach.

We employ ResNet50 (pre-trained on ImageNet) for water detection. Images are resized to the network's input size (227x227) while keeping the original aspect ratio. Horizontal flipping, brightness variations and non-uniform re-scaling of the images are applied for data augmentation. The top five layers are fine-tuned (6 epochs, batch size 256) before the whole network is trained using Adam optimizer (10 epochs, batch size 32, learning rate 10^{-4}). We employ OpenPose [3] to detect body joints from depicted human bodies. To filter out unreliable skeletons, we exclude those with a confidence score (C_U) - calculated from the two most robust upper body parts (head and chest) - below an empirically estimated threshold of 0.6. We calculate a mean confidence score (C_L) over the lower body parts (knees and feet). To determine whether the lower extremi-

ties of a skeleton are visible, we employ the following heuristic rule: $C_U / \max(C_L, 10^{-4}) > T$, with T being an empirically determined threshold of 1.5. Finally, positive predictions of the rule-based classifier and the water detector implies a positive detection of a person standing in knee-high water.

3. Datasets

Experiments are carried out on two datasets provided by the MediaEval Benchmark Multimedia Satellite Task 2018 (MMSat18) and 2019 (MMSat19) respectively [1, 2]. All available data is manually annotated (water/no water) and used to train the water detector. A total of 13.761 image annotations (5.395 water, 8.366 no water) along with corresponding image URLs (incl. download tool) as well as our ResNet50 model weights can be accessed publicly[1].

4. Results & Discussion

For experimental evaluations, we randomly split the MMSat19 data into training (80%) and validation (20%) sets preserving class priors. Testing is performed on the (non-public) test set of the MMSat19 benchmark. For the global approach (GA), we used the pipeline in Figure 1A. First, the water detector is trained only on the MMSat19 data (GA-1) and later on both datasets (GA-2). For the local approach (LA), we used the pipeline in Figure 1B with MMSat19 data. Finally, we apply majority voting to all three approaches.

Due to the imbalanced data, we used macro averaged F1-scores as performance measure. The experimental results surpass the random baseline of 0.5, which shows that our models are able to learn useful patterns. The results on the test set show only minor differences between the four approaches. The overall performance is similar on the validation and test sets, which indicates a good generalization ability. The classification accuracy of the water detector is quite high with 88% (not shown in Table 1). The main source of failure are false detections of the pose tracker due to occlusions by foreground objects and reflections in the water (see Figure 2). Potential improvements identified include the use of several pose estimators trained on content from different environments, e.g., rural and urban areas. Additionally, pixel-wise classification (segmentation) of water and human bodies could be useful to deal with occlusions and reflection in the water.

[1] https://tinyurl.com/waterDetectionDataset



Figure 2: Challenges: misleading images (left), water reflections (middle) and occlusions (right).

Approach	Validation (P/R/F1)			Test (F1)
GA-1	0.58	0.67	0.61	0.61
GA-2	0.55	0.60	0.56	0.59
LA	0.58	0.77	0.60	0.59
Majority Voting	0.59	0.68	0.61	0.61

Table 1: Macro-averaged precision (P), recall (R), and f1-scores for visual flood level estimation.

5. Conclusion

Our experiments show that pose estimation and water detection provide useful clues for the assessment of flood levels. By building upon skeletons, the presented approach is invariant to gender, age and height. Main challenges for robust water level estimation represent occlusions and reflections. For future work, a larger, more balanced and more heterogeneous dataset is needed.

Acknowledgments

This work was supported by the Austrian Research Promotion Agency (FFG), Project nos. 855784, 856333, and 865973.

References

[1] B. Bischke, P. Helber, S. Brugman, E. Basar, Z. Zhao, and M. Larson. The multimedia satellite task at MediaEval 2019: Estimation of flood severity. In *Working Notes Proc. of MediaEval Wshp. (to appear)*.

[2] B. Bischke, P. Helber, Z. Zhao, J. de Bruijn, and D. Borth. The multimedia satellite task at MediaEval 2018: Emergency response for flooding events. In *Working Notes Proc. of the MediaEval Wshp.*, 2018.

[3] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. Realtime multi-person 2D pose estimation using part affinity fields. In *Proc. of CVPR*, 2017.

[4] M. Larson, P. Arora, C.-H. Demarty, M. Riegler, B. Bischke, E. Dellandrea, M. Lux, A. Porter, and G. J. Jones. *"Working Notes Proc. of the MediaEval 2019 Wshp. (to appear)"*.

[5] M. Zaffroni, L. Lopez-Fuentes, A. Farasin, P. Garza, and H. Skinnemoen. AI-based flood event understanding and quantification using online media and satellite data. In *"Working Notes Proc. of the MediaEval 2019 Wshp. (to appear)"*.

Real-World Video Restoration using Noise2Noise

Martin Zach, Erich Kobler
Institute of Computer Graphics and Vision
{martin.zach@student, erich.kobler@icg}.tugraz.at

Abstract. *Restoration of real-world analog video is a challenging task due to the presence of very heterogeneous defects. These defects are hard to model, such that creating training data synthetically is infeasible and instead time-consuming manual editing is required. In this work we explore whether reasonable restoration models can be learned from data without explicitly modeling the defects or manual editing. We adopt Noise2Noise techniques, which eliminate the need for ground truth targets by replacing them with corrupted instances. To compensate for temporal mismatches between the frames and ensure meaningful training, we apply motion correction. Our experiments show that video restoration can be learned using only corrupted frames, with performance exceeding that of conventional learning.*

1. Introduction

Recently the approach to signal reconstruction from corrupted measurements shifted from explicitly modeling the statistics of the corruptions and image priors, *e.g*. Block-matching and 3D filtering (BM3D) [6] or Total Variation (TV) based methods [4, 24], to learning based techniques such as Convolutional Neural Networks (CNNs) [11]. Since then, deep learning techniques [9, 18] have become very popular. Residual learning [9], batch normalization [10] and similar improvements along with increasing computational power and high quality datasets made it possible to train such architectures efficiently. Deep architectures are now the state-of-the-art for many image restoration tasks such as denoising, deblurring, and inpainting [8, 13, 19] as well as semantic segmentation [16, 23] and classification [27].

Despite these advances, generalization performance of such models is still largely limited by the size of the available dataset. The acquisition of clean targets is often very tedious or difficult and it has been proposed that data collection is becoming the critical bottleneck in machine learning [22]. It is therefor interesting to investigate whether networks can learn meaningful mappings when only being presented corrupted samples — both as input and as target. Lethinen *et al*. [15] showed that clean targets are not required to learn meaningful reconstructions, provided that the corrupted samples are drawn from an arbitrary distribution conditioned on the clean target which needs to be the expected value. This technique now known as Noise2Noise (N2N) has been successfully applied to image restoration tasks [14].



Figure 1. Sample from the dataset, corrupted by typical temporally incoherent and very local defects highlighted in orange.

In this work we explore the applicability of N2N for video denoising, especially concerning the real-world case of having finite data. Due to the nature of the defects, acquiring ground truth samples would require manual editing of the frames and is often not feasible. Further, the defects are very complex and divers in nature such that modeling them is difficult to impossible. Figure 1 displays such an example, where temporally incoherent defects with small spatial extent and high inter-pixel correlation can be seen.

The N2N setting imposes limitations that require special considerations. Since different frames show the scene at different points in time, they cannot directly be used as training pairs. We overcome this by separating temporal motion compensation and spatial denoising, allowing corrupted samples to be both in-

put and target for the model. With this architecture we were able to achieve satisfactory results, showing that video restoration can be done entirely without ground truth data. This significantly eases the task by avoiding the requirement for tedious manual labeling.

2. Related Work

Learning-based Image Restoration Convolutional Neural Networks (CNNs) were first used in 2008 [11], where they achieved similar performance to model based approaches. Later, Burger *et al*. [2] showed that shallow plain Multi Layer Perceptrons (MLP) can achieve results comparable to BM3D. The DnCNN [30] combined recent advances such as the convolutional structure, global residual learning [27], batch normalization [10], and a ReLU activation [20] to achieve a significant performance increase over state-of-the-art explicit models. Later, the FFDNet [31] extended the DnCNN by the use of input noise maps to account for spatially varying noise intensity, in order to apply it to real-world photographs. CBDNet [8] builds on this idea and introduces a noise estimation subnetwork whose output is fed into the denoising network along with the image to achieve notably good results for real-world denoising.

Video Restoration Compared to image denoising, little work exists on video denoising. Patch-based approaches are still the most prominent, *e.g*. V-BM4D [17] and Video Non-Local Bayes (VNLB) [1]. The Deep Video Denoising Network (DVDNet) [28] was one of the first convolutional network approaches to outperform VNLB, whilst being computationally more efficient. In the DVDNet, two separate networks are used for spatial and temporal denoising, and adjacent frames are motion compensated using DeepFlow [29]. Similarly, ViDeNN [5] uses separated spatial and temporal denoising networks, but motion compensation is learned in the temporal network. Frame-to-frame Training [7] exploits N2N by fine-tuning a pretrained network on motion-compensated successive frames. However, the applicability to real-world data remains limited since only one frame is considered for restoration. Besides denoising, learning based methods have been successfully applied to frame interpolation [21], super resolution [3] and deblurring [25].

3. Methods

We consider video scenes $\xi_i = (x_j^i)_{j=1}^{N_f}$ consisting of N_f frames $x_j^i \in \mathbb{R}^{n3}$ with a resolution $n = n_1 \times n_2$ and RGB channels. Each frame of a scene x_j^i is assumed to be corrupted by additive noise, i.e.

$$x_j^i = y_j^i + n_g + n_d \,, \tag{1}$$

where y_j^i is the underlying clean true frame, n_g models noise due to film grain and n_d represents the spatially correlated single-frame defects highlighted in Figure 1. Both noise sources are uncorrelated across the temporal dimension due to the stochastic nature of film grain n_g and the temporal incoherence of n_d. We note that the approach is not limited to this noise model.

3.1. Models for Single-Frame Defect Restoration

The simplest approach to estimate the clean true frame y_j^i is by means of single-frame denoising. For this setting we use the DnCNN [30] to generate a prediction $\hat{y}_j^i$ by

$$\hat{y}_i^j = \mathcal{N}_S^\theta(x_j^i) \,, \tag{2}$$

solely based on the single corresponding corrupted frame x_j^i. Here, θ are the parameters of the DnCNN. They are learned from data either by supervised learning (SL) — provided that target frames are available — or by the N2N approach, which we describe later in this section. The major disadvantage of the single-frame denoising approach is that the model cannot exploit temporal information to detect and restore the single-frame defects.

To overcome this issue and enable the extraction of temporal features, we propose to learn a variant of the DnCNN model operating on two consecutive frames. These two adjacent frames need to be aligned to compensate the motion in dynamic scenes and ease the denoising problem. In detail, we account for the motion by computing the optical flow

$$f_{zj}^i = \mathcal{F}(x_z^i, x_j^i) \tag{3}$$

from frame x_z^i to x_j^i, where $\mathcal{F}\colon \mathbb{R}^{n3} \times \mathbb{R}^{n3} \to \mathbb{R}^{n2}$ implements the pretrained PWC-Net [26]. Using the thereby estimated flow f_{zj}^i, we warp a frame x_z^i of the scene onto the reference frame x_j^i by

$$\hat{x}_{zj}^i = \mathcal{W}(x_z^i, f_{zj}^i) \tag{4}$$

to obtain the motion compensated frame $\hat{x}^i_{zj}$, where $\mathcal{W}\colon \mathbb{R}^{n3} \times \mathbb{R}^{n2} \to \mathbb{R}^{n3}$ is the bilinear warping operator.

In addition, we also compute the backward flow f^i_{jz} and perform a forward-backward check to obtain a binary mask $m^i_{zj} \in \{0,1\}^n$ in the reference frame x^i_j discarding occluded areas. To enable an effective detection of the single-frame defects using temporal information, we require the flow estimation to interpolate over the defects such that they are considered valid in the mask.

Combining the motion compensated frame and the mask with the reference frame x^i_j yields the input to the dynamic model $\mathcal{N}^\theta_D\colon \mathbb{R}^{n3}\times\mathbb{R}^{n3}\times\{0,1\}^n \to \mathbb{R}^{n3}$. Its output

$$\hat{y}^i_{zj} = \mathcal{N}^\theta_D(x^i_j, \hat{x}^i_{zj}, m^i_{zj}) \tag{5}$$

is the estimation of the clean true frame combining spatial and temporal information from two adjacent frames. As before θ denotes the trainable parameters of the DnCNN model learned from data by a SL or N2N approach.

3.2. Supervised and Noise2Noise Learning

Let us first consider supervised learning for reconstructing single-frame defects. Here one requires for every training sample frame x^i_j a corresponding target frame $\bar{y}^i_j$, which can be created by tedious and time-consuming manual editing. Given a collection of corrupted video scenes $\{\xi_i = (x^i_1, \dots, x^i_{N_f})\}_{i=1}^{N_s}$ and a corresponding manually edited target scene $\{\psi_i = (\bar{y}^i_1, \dots, \bar{y}^i_{N_f})\}_{i=1}^{N_s}$, we define the supervised training problem as

$$\min_\theta \sum_{i=1}^{N_s} \mathcal{L}^{\mathrm{SL}}_{\{\mathrm{S,D}\}}(\xi_i, \psi_i, \theta)\,. \tag{6}$$

The scene specific loss $\mathcal{L}^{\mathrm{SL}}_{\{\mathrm{S,D}\}}$ depends on the considered model. For the static model $\mathcal{N}^\theta_S$ we use

$$\mathcal{L}^{\mathrm{SL}}_{\mathrm{S}}(\xi_i, \psi_i, \theta) = \sum_{j=1}^{N_f} \ell\left(\mathcal{N}^\theta_S(x^i_j) - \bar{y}^i_j\right)\,, \tag{7}$$

whereas the loss for the dynamic model $\mathcal{N}^\theta_D$ is given by

$$\mathcal{L}^{\mathrm{SL}}_{\mathrm{D}}(\xi_i, \psi_i, \theta) = \sum_{j=1}^{N_f}\sum_{\substack{z=1\\z\neq j}}^{N_f} \ell\left(\mathcal{N}^\theta_D(x^i_j, \hat{x}^i_{zj}, m^i_{zj}) - \bar{y}^i_j\right)\,, \tag{8}$$

where $\ell \in \{\|\cdot\|_1, \|\cdot\|_2^2, \|\cdot\|_\epsilon\}$ and $\|x\|_\epsilon = \sum_i |x_i|_\epsilon$ is the Huber norm using

$$|x|_\epsilon = \begin{cases} \frac{1}{2}x^2 & \text{if } |x| \le \epsilon \\ \epsilon(|x| - \frac{1}{2}\epsilon) & \text{else} \end{cases}\,. \tag{9}$$

Despite the constant number of training sample frames, we can use $N_s N_f(N_f-1)$ pairs for training the dynamic model due to the possible permutations, a factor of (N_f-1) more than for the static model.

To avoid the manual editing of target frames, we propose to adopt the N2N approach to remove single-frame defects. Thus, only the corrupted video scenes $\{\xi_i = (x^i_1, \dots, x^i_{N_f})\}_{i=1}^{N_s}$ are used during training. We modify the training problem for N2N to estimate the learnable parameters θ of the models to

$$\min_\theta \sum_{i=1}^{N_s} \mathcal{L}^{\mathrm{N2N}}_{\{\mathrm{S,D}\}}(\xi_i, \theta)\,, \tag{10}$$

using the specific scene loss for the static model

$$\mathcal{L}^{\mathrm{N2N}}_{\mathrm{S}}(\xi_i, \theta) = \sum_{j=1}^{N_f}\sum_{\substack{k=1\\k\neq j}}^{N_f} \ell\left(m^i_{kj} \odot (\mathcal{N}^\theta_S(x^i_j) - \hat{x}^i_{kj})\right) \tag{11}$$

and for the dynamic model

$$\mathcal{L}^{\mathrm{N2N}}_{\mathrm{D}}(\xi_i, \theta) = \sum_{j=1}^{N_f}\sum_{\substack{z=1\\z\neq j}}^{N_f}\sum_{\substack{k=1\\k\neq j\\k\neq j}}^{N_f} \ell\left(m^i_{kj} \odot (\mathcal{N}^\theta_D(x^i_j, \hat{x}^i_{zj}, m^i_{zj}) - \hat{x}^i_{kj})\right)\,. \tag{12}$$

This is illustrated in Figure 2. In contrast to supervised learning, we choose a frame x^i_k and compensate for the motion to the reference frame x^i_j and get the warped frame $\hat{x}^i_{kj}$ as well as the binary mask m^i_{kj}. Then we only evaluate the loss function in the areas where the forward-backward check is consistent to disregard motion estimation errors. A particular advantage of N2N learning is that a factor of (N_f-1) more training samples are available for the static model and (N_f-2) for the dynamic model without the necessity to manually edit any frame.

In all our numerical experiments we optimize (6) and (10) using a dataset of $N_s = 368$ video sequences of $N_f = 3$ frames, which was divided into training (343) and test set (25). For each of the 368 samples there is 1 manually edited target at $j = 2$, where only the single-frame defects n_d were removed and the film

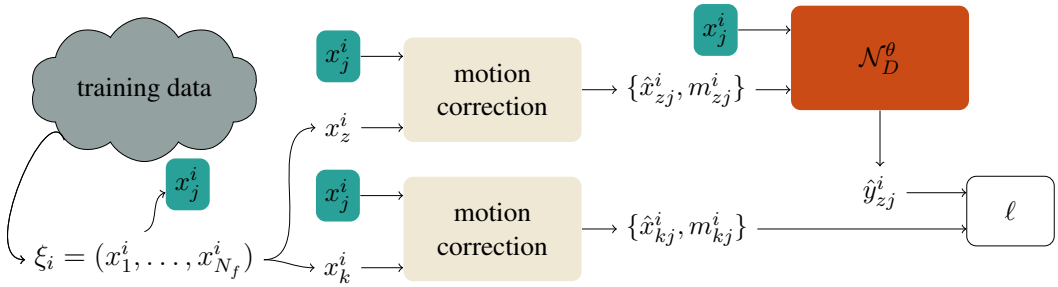


Figure 2. Illustration of the proposed sampling process for N2N learning to video restoration using motion compensation. Here we choose x_j^i as the reference frame, and warp x_z^i and x_k^i onto it. Then, we calculate the estimate $\hat{y}_{zj}^i$ by using the reference frame x_j^i and $\hat{x}_{zj}^i$, and finally the loss using $\hat{x}_{kj}^i$.

Error	static		dynamic	
	SL	N2N	SL	N2N
ℓ_2	0.002 151	0.018 161	0.000 675	0.001 648
ℓ_1	0.002 736	0.012 005	0.000 320	0.001 910
$\epsilon = 0.1$	—	—	0.000 721	0.001 630

Table 1. Evaluation of the average mean squared error to the manually edited target images of the test set.

	Original	SL	N2N
Overall Best	3.13 %	43.23 %	53.65 %
Least Flickering	0.52 %	10.94 %	88.54 %
Significant Smoothing	0 %	1.04 %	56.77 %

Table 2. Quantitative evaluation of the reader study. The results of indicate that the majority of participants prefers the N2N method, where artifacts are significantly better removed at the cost of introducing some smoothing.

grain was not changed. We used a pre-trained PWC-Net [26] for motion compensation and extended the DnCNN [30] to 20 layers with batch normalization, and 64 convolution kernels of size 3×3. Using the ADAM [12] optimizer on a batch size of 128, we trained the models for 3000 iterations with a learning rate of $\alpha = 1 \times 10^{-4}$ and decay rates of $\beta_1 = 0.9$ and $\beta_2 = 0.999$. We sampled patches of size 64×64 from the frames and augmented the data by vertical and horizontal flipping. Finally, we estimate $\hat{y}_2^i$ as

$$\hat{y}_2^i = \begin{cases} \hat{y}_{12}^i & \text{if } m_{12}^i \wedge (\neg m_{32}^i) \\ \hat{y}_{32}^i & \text{if } m_{32}^i \wedge (\neg m_{12}^i) \\ \frac{\hat{y}_{12}^i + \hat{y}_{32}^i}{2} & \text{else} \end{cases} . \tag{13}$$

4. Results

In this section we present results to highlight the benefits of N2N learning for removing single-frame defects in scanned historical video scenes. We perform quantitative and qualitative evaluation for the static and dynamic models and compare supervised learning to N2N. The qualitative results were also evaluated in a reader study with a focus on temporal coherence.

We show the Mean Squared Error (MSE) on the test set in Table 1 and some representative examples in Figure 3. Given the nature of the defects, their detection is easier if the model can use temporal information. This is confirmed by the results in Table 1, since the results show that the dynamic model outperforms the static model.

The numerical results indicate better performance for the models trained on SL targets. However, this is misleading since it does not necessarily correspond to better defect removal. In fact, Figure 3 suggests that N2N learning improves defect removal. The superior MSE of supervised models is explained by the preservation of film grain, which has not been removed in the targets. In contrast, since film grain differs between the frames, N2N models learn to remove it. Thus, even though they are qualitatively better at removing defects, they yield worse numerical errors.

Further, visual quality of videos cannot be determined by considering the individual frames only. The temporal context needs to be considered as well, where incoherencies can lead to an unpleasant viewing experience. Quality measures could be improved by taking temporal coherency into account, however objective evaluation would still be problematic. Thus, numerical error measures are not suited to fully determine the visual quality of the output.

In general, evaluation is best done by a human who can subjectively decide whether, *e.g.*, removal of film grain is desired, and how pleasant the final video is to watch over all. We therefor conducted a reader study[1]

[1]Material available at https://github.com/zacmar/restoration-reader-study

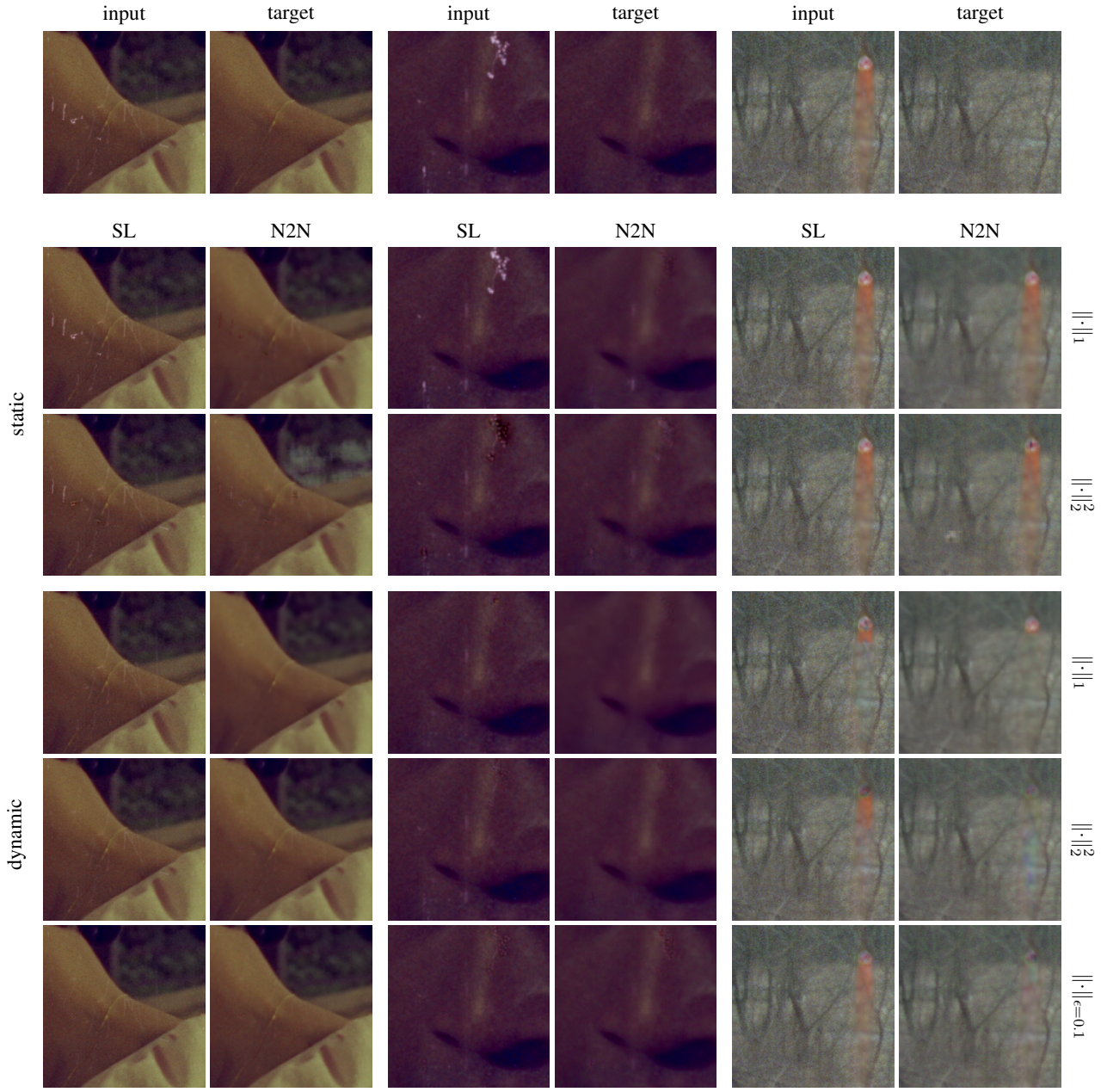


Figure 3. The first row depicts crops from the corrupted frame x_j^i along with the corresponding manually edited target $\bar{y}_j^i$. The second and third row show the results obtained using the static model $\mathcal{N}_S^\theta$, whereas, the results of the dynamic model are depicted in the last three rows. The columns alternate between supervised learning (SL) and N2N results and on the right we show which loss function was used during training.

in which the reader was presented three versions of the same scene side by side: (i) The original frames, the output of the models trained using (ii) SL and (iii) N2N ($\|\cdot\|_\epsilon, \epsilon = 0.1$). Table 2 presents the results obtained from 24 people who were each shown 8 video sequences. It shows that the model trained with N2N is best at removing the defects, at the cost of over smoothing the images. Still, it was the overall preferred method, with $53.65\,\%$ of all samples being deemed "Overall Best" by the participants.

5. Conclusion

In this work we explored the possibilities of using N2N learning for video restoration. We trained static and dynamical models by considering adjacent frames using supervised learning and N2N, relying on robust motion estimation. Using this paradigm we demonstrated that video restoration can be learned by only looking at corrupted frames at performance levels exceeding those of supervised learning. This opens

up new possibilities in areas where acquiring clean training data is too time consuming or infeasible.

There are some limitations that we leave for future research. Due to the structure of our dataset, the number of samples available for N2N learning was limited by the available ground truth targets. Since N2N does not require manual frame editing, it is possible to increase the size of the dataset without much effort. Along with the increase of the size of the dataset, the model complexity could be increased, typically resulting in better performance.

Acknowledgements

The authors acknowledge grant support from the National Institutes of Health under grant 1R01EB024532-02.

References

[1] P. Arias and J. Morel. Video denoising via empirical bayesian estimation of space-time patches. *JMIV*, 60:70–93, 2018.

[2] H. C. Burger, C. J. Schuler, and S. Harmeling. Image denoising: Can plain neural networks compete with BM3D? In *CVPR*, 2012.

[3] J. Caballero, C. Ledig, A. Aitken, A. Acosta, J. Totz, Z. Wang, and W. Shi. Real-time video super-resolution with spatio-temporal networks and motion compensation. *CVPR*, 2017.

[4] A. Chambolle. An algorithm for total variation minimization and applications. *JMIV*, 20(1):89–97, 2004.

[5] M. Claus and J. C. van Gemert. ViDeNN: Deep blind video denoising. In *CVPR Workshops*, 2019.

[6] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IP*, 16(8):2080–2095, 2007.

[7] T. Ehret, A. Davy, J. Morel, G. Facciolo, and P. Arias. Model-blind video denoising via frame-to-frame training. In *CVPR*, 2019.

[8] S. Guo, Z. Yan, K. Zhang, W. Zuo, and L. Zhang. Toward convolutional blind denoising of real photographs. In *CVPR*, 2019.

[9] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[10] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

[11] V. Jain and H. Seung. Natural image denoising with convolutional networks. In *NIPS*, 2008.

[12] D. P. Kingma and J. L. Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[13] E. Kobler, T. Klatzer, K. Hammernik, and T. Pock. Variational networks: Connecting variational methods and deep learning. In *GCPR*, 2017.

[14] S. Laine, J. Lehtinen, and T. Aila. Improved self-supervised deep image denoising. In *ICLR*, 2019.

[15] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, and T. Aila. Noise2Noise: Learning image restoration without clean data. In *ICML*, 2018.

[16] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

[17] M. Maggioni, G. Boracchi, A. Foi, and K. Egiazarian. Video denoising, deblocking, and enhancement through separable 4-d nonlocal spatiotemporal transforms. *IP*, 21(9):3952–3966, 2012.

[18] X.-J. Mao, C. Shen, and Y.-B. Yang. Image restoration using convolutional auto-encoders with symmetric skip connections. *ArXiv*, abs/1606.08921, 2016.

[19] X.-J. Mao, C. Shen, and Y.-B. Yang. Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In *NIPS*, 2016.

[20] V. Nair and G. Hinton. Relus improve restricted boltzmann machines. In *ICML*, 2010.

[21] S. Niklaus and F. Liu. Context-aware synthesis for video frame interpolation. In *CVPR*, 2018.

[22] Y. Roh, G. Heo, and S. E. Whang. A survey on data collection for machine learning: A big data - ai integration perspective. *KDE*, 2019.

[23] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

[24] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60(1):259 – 268, 1992.

[25] S. Su, M. Delbracio, J. Wang, G. Sapiro, W. Heidrich, and O. Wang. Deep video deblurring for hand-held cameras. *CVPR*, 2017.

[26] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz. PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. In *CVPR*, 2018.

[27] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2015.

[28] M. Tassano, J. Delon, and T. Veit. DVDnet: A fast Network for Deep Video Denoising. In *ICIP*, 2019.

[29] P. Weinzaepfel, J. Revaud, Z. Harchaoui, and C. Schmid. Deepflow: Large displacement optical flow with deep matching. *ICCV*, 2013.

[30] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang. Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising. *IP*, 26(7):3142–3155, 2017.

[31] K. Zhang, W. Zuo, and L. Zhang. Ffdnet: Toward a fast and flexible solution for CNN based image denoising. *IP*, 2018.

Asymptotic Analysis of Bivariate Half-Space Median Filtering

Martin Welk
UMIT – Private University for Health Sciences, Medical Informatics and Technology
Hall/Tyrol, Austria
martin.welk@umit.at

Abstract. *Median filtering is well established in signal and image processing as an efficient and robust denoising filter with favourable edge-preserving properties, and capable of denoising some types of heavy-tailed noise such as impulse noise. For multi-channel images such as colour images, flow fields or diffusion tensor fields, multivariate median filters have been considered in the literature. Whereas the L^1 median filter so far dominates in image processing applications, other multivariate concepts from statistics may be used such as the half-space median which in the focus of this work.*

In the understanding of discrete image filters a central question is always how these relate to the space-continuous physical reality underlying discrete images. For the univariate median filter, a milestone in answering this question is an asymptotic approximation result that links median filtering to the mean curvature motion evolution. We will present an analogous result for half-space median filtering in the bivariate (two-channel) case, which contributes to the theoretical understanding of multivariate median filtering and provides the basis for further generalisations in future work.

1. Introduction

Median filtering [10] is a well-established procedure in signal and image processing. For grey-value images it is known as an efficient and robust denoising method with favourable edge-preserving properties. In standard median filtering, a pixel mask (for example, a $(2m+1) \times (2m+1)$ square, or a discrete approximation of a disc) is moved as a sliding window across the image. At each pixel location, the mask is used to select grey-values of the input image; the median of these grey-values is then assigned to the central pixel as its new grey-value in the output image. This filter can also be iterated, which is then called iterated median filtering.

Continuous median filtering. Thus, median filtering is designed in the first place as a discrete procedure. An important question regarding its validity for images is therefore whether it is in a sound relationship to the underlying continuous nature of images. This is indeed the case: Firstly, it is straightforward to conceive mathematically a median filter for space-continuous images: Given an image as a function over a planar domain, one can cut out a neighbourhood around each location in the plane (say, a square or disc centered at the reference point) and determine the median of the (continuous) distribution of image values within this neighbourhood. Discrete median filtering of a sampled image approximates this concept. Secondly, assuming disc-shaped neighbourhoods (of radius ϱ) in this process, it has been proven in [5] that iterated space-continuous median filtering approximates a partial differential equation (PDE) as $\varrho \to 0$ in the sense that one space-continuous median filter step asymptotically approximates a time step of size $\varrho^2/6$ of an explicit time discretisation of the mean curvature motion PDE $u_t = |\nabla u| \operatorname{div}\bigl(\nabla u/|\nabla u|\bigr)$ for the planar image u evolving in time.

Multivariate medians. Due to the success of median filtering for grey-value images, researchers have proposed generalisations of the median filter to multi-channel images (such as colour images, optic flow fields, diffusion tensor fields). After early attempts such as the *vector median filters* from [1] which focussed on methods to select one vector from a given set of input vectors as its median, attention turned soon to multivariate median concepts known from the statistical literature in which the median



Figure 1. From left to right: Synthetic test image (30×30 pixels) in orange–blue colour space. – Componentwise median filtering. – L^1 median filtering. – Oja median filtering. – Half-space median filtering. For all median filters the sliding window was a discrete disc of radius $\sqrt{5}$, and one iteration was applied.

of multivariate data (such as points in the plane or space) is not restricted to be one of the input data. The L^1 *median* [12] was the first concept of this kind discussed in the statistical literature [2, 4, 13] and also in image processing [9, 17]. Shortcomings of this concept, especially its lack of affine equivariance which contrasts to the very general monotonous equivariance of the classical univariate median, led statisticians to alternative concepts such as Oja median [7], half-space median [6, 11] and convex-hull-stripping median [3, 8].

All of these multivariate medians are defined in the first place as discrete concepts: Given a set of points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m$ in $\mathbb{R}^n$, they yield a median $\boldsymbol{\mu} \in \mathbb{R}^n$. Algorithmically, their application to multivariate images is straightforward; however, the validity of such a procedure again depends on the question whether it approximates a suitable filter for space-continuous images. Furthermore, the question arises whether a PDE can be stated that is approximated by such a space-continuous multivariate median filter. For the L^1 median and Oja median, these questions have been answered in [14]: The definition of space-continuous variants of these filters is more or less straightforward, and PDE limits could be stated for images with values in $\mathbb{R}^2$ and $\mathbb{R}^3$. For the half-space median, a space-continuous counterpart has been described in [15] but the PDE limit (in $\mathbb{R}^2$) was stated only as a conjecture, without proof. We mention that for the convex-hull-stripping median stating a space-continuous filtering procedure is already a difficult task in itself, see [16].

Our contribution. The purpose of this work is to advance the theoretical understanding of half-space median filtering as a multivariate image filter. We will derive the PDE approximated by space-continuous half-space median filtering of bivariate images, thereby proving the conjecture stated in [15]. Aspects of practical application are not in the foreground at the present stage of research; examples are presented just for illustrating the properties of multivariate median filters, and are restricted to the bivariate case (notwithstanding the greater practical importance of three-channel colour images).

Structure of the paper. After shortly demonstrating the effect of multivariate median filters, we will recall the definition of the half-space median for discrete data and its space-continuous analogue in Section 2. In Section 3 we will prove the PDE approximation result as conjectured in [15]. A short summary and outlook in Section 4 concludes the paper.

2. Multivariate Median Filtering

The univariate median filter excels as an edge-preserving denoising filter for images that can deal well with types of noise such as impulse noise. Unfortunately, for multi-channel images a straightforward generalisation by using the median just for each channel separately does not lead to reasonable results as we demonstrate by a small synthetic example in Figure 1. For simplicity, and since our theoretical work presented in the next section is currently restricted to the bivariate case, we use a test image with just two colour channels (yellow and blue) which is degraded by pepper noise (impulsive noise consisting of black noise pixels). Whereas componentwise median filtering removes noise pixels in homogeneous colour regions, it even amplifies noise near colour edges. A more plausible filtering result is achieved by multivariate median filters three of which are demonstrated in the figure: the L^1 median filter (see e.g. [9]), the Oja median filter (see e.g. [14]) and the half-space median filter which is in the focus of the present paper. As can be seen, the multivariate median filters lead to some interpolation between the two colours near edges but don't

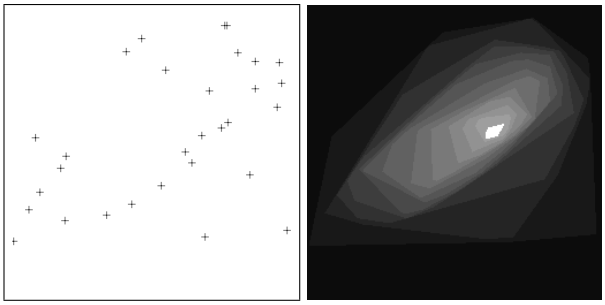


Figure 2. Left: Set of 30 sample points in the plane. Right: Map of half-space depths w.r.t. the sample points. Points in the white area are half-space medians.



Figure 3. Left: Test image *sailboat* (512×512 pixels) reduced to yellow–blue colour space. Right: Half-space median filtering result, using a discrete disc of radius 2 as sliding window, 5 iterations.

amplify noise. Whereas the L^1 median filter yields the visually most appealing result in this example, its underlying median concept relies on Euclidean distances which might not be always be meaningful in applications. The Oja simplex median as well as the half-space median cure this weakness (they are affine equivariant), with the half-space median yielding a better denoising result in this example.

Discrete half-space median. Let us shortly recall the definition of half-space median based on [6, 11]. Given points $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_m \in \mathbb{R}^n$, the *half-space depth* of a point $\boldsymbol{p} \in \mathbb{R}^n$ is the minimal number of data points that can lie on one side of a hyperplane through $\boldsymbol{p}$. For example, the half-space depth of any $\boldsymbol{p}$ outside the convex hull of the given points is zero because there exists a hyperplane through $\boldsymbol{p}$ which does not split the data points at all. In contrast, if there is a $\boldsymbol{p}$ somewhere in the middle of the given points for which any hyperplane through $\boldsymbol{p}$ splits the data set in half, it will have a half-space depth equal or close to $m/2$. A half-space median of the given data is then simply a point of maximal half-space depth, see Fig. 2 for an example. For discrete data sets, there is in general a convex polyhedron in $\mathbb{R}^n$ consisting entirely of half-space medians. We will not further discuss this underdetermination, however, as it plays no role in the continuous situation.

Application of the discrete half-space median for the filtering of $\mathbb{R}^n$-valued images is in principle straightforward: A sliding window is used to select at each pixel location a set of neighbouring pixels, and the half-space median of their values becomes the new image value at the given pixel. Practically, however, the algorithmic complexity of the half-space median computation is an issue which requires further work, see the remarks in [15].

Having shown a synthetic example in Figure 1, we present the result of half-space median filtering on a natural colour image (reduced to two colour channels) in Figure 3. Similar to the classical median filter for grey-value images, the iterated multivariate median filter removes small details and simplifies contours. Notice, however, that a slight blurring of edges occurs, albeit much less than in linear filters such as box averaging (with the same window size as in the median filter) or Gaussian smoothing (with a comparable standard deviation).

Continuous half-space median. In a continuous setting, the discrete set of data points is replaced with a density over $\mathbb{R}^n$, i.e., an integrable function γ with total weight 1. The half-space depth of $\boldsymbol{p} \in \mathbb{R}^n$ then is the minimum among all integrals of γ over half-spaces cut off by hyperplanes through $\boldsymbol{p}$. Again, the half-space median of γ is the point of maximal half-space density, which will be unique in generic cases.

The construction of a half-space median filter for space-continuous $\mathbb{R}^n$-valued images is again a straightforward adaptation of the univariate procedure, with the density of image values within a sliding neighbourhood of each image location being the input from which the continuous half-space median is taken.

Affine equivariance. The definitions of half-space depths and and half-space medians rely only on incidence relations between points and half-spaces in the data space. Affine transforms of the data space preserve all of these relations. As a consequence, for any such affine transform the half-space median of the transformed input data coincides with the transformed half-space median of the original data. This is dubbed by saying that the half-space median is *affine equivariant*. This property ensures that the

half-space median can be applied meaningfully to data for which no physically meaningful Euclidean structure in $\mathbb{R}^n$ can be assumed (e.g., if different dimensions of the data space refer to incommensurable physical quantities).

3. PDE Limit of Half-Space Median Filtering

Our main theoretical result is the following proposition which was already stated as a conjecture in [15]. It specifies the space- and time-continuous image evolution which is approximated by iterated space-continuous half-space median filtering in the limit case when the radius of the sliding window goes to zero, thereby generalising the result from [5] for the univariate median filter and results from [14] for other multivariate median filters. In particular, the approximated PDE is identical with the one approximated by the Oja median filter, see [14, 15].

Proposition 1 *Let $\boldsymbol{u} : \mathbb{R}^2 \supset \Omega \to \mathbb{R}^2$, $(x, y) \mapsto (u, v)$ be a smooth bivariate image over a compact domain Ω. At any regular location $\boldsymbol{x} = (x, y) \in \Omega$, i.e., for which the Jacobian $\mathrm{D}\boldsymbol{u}(\boldsymbol{x})$ is of rank 2, one step of space-continuous half-space median filtering with a disc-shaped window of radius ϱ approximates a time step of an explicit time discretisation of the PDE*

$$\boldsymbol{u}_t = 2\,\Delta\boldsymbol{u} + \boldsymbol{A}\,(\boldsymbol{u}_{yy} - \boldsymbol{u}_{xx}) + \boldsymbol{B}\,\boldsymbol{u}_{xy} \qquad (1)$$

with time step size $\varrho^2/24$, where the coefficient matrices $\boldsymbol{A} \equiv \boldsymbol{A}(\mathrm{D}\boldsymbol{u})$, $\boldsymbol{B} \equiv \boldsymbol{B}(\mathrm{D}\boldsymbol{u})$ are given by

$$\boldsymbol{A} = \frac{1}{u_x v_y - u_y v_x} \begin{pmatrix} u_x v_y + u_y v_x & -2u_x u_y \\ 2v_x v_y & -u_x v_y - u_y v_x \end{pmatrix}, \quad (2)$$

$$\boldsymbol{B} = \frac{2}{u_x v_y - u_y v_x} \begin{pmatrix} u_x v_x - u_y v_y & -u_x^2 + u_y^2 \\ v_x^2 - v_y^2 & -u_x v_x + u_y v_y \end{pmatrix}. \quad (3)$$

The proof of this result relies on the following lemma.

Lemma 2 *Let $\mathbf{u}$ be as in Proposition 1, and let $\boldsymbol{x}_0 = \mathbf{0} \in \Omega$ be a regular point for which $\boldsymbol{u}(\boldsymbol{x}_0) = \mathbf{0}$, and $\mathrm{D}\boldsymbol{u}(\boldsymbol{x}_0)$ is the 2×2 unit matrix. Then one step of space-continuous half-space median filtering with a disc-shaped window of radius ϱ approximates at $\boldsymbol{x}_0$ a time step of an explicit time discretisation of the PDE system*

$$u_t = u_{xx} + 3u_{yy} - 2v_{xy}\,, \qquad (4)$$

$$v_t = 3v_{xx} + v_{yy} - 2u_{xy} \qquad (5)$$

with time step size $\varrho^2/24$.

Note that the lemma states the approximation result of the proposition for a specific geometric configuration where the gradients of the components u, v of $\boldsymbol{u}$ are locally aligned with the x, y coordinate axes and of unit magnitude. This special geometric situation also helps in understanding the effect of the PDE of the proposition. A more detailed discussion is found in [14, Sect. 3.1.3] from which we shortly recall the main facts. First, the right-hand side contains terms which play a similar role as the mean curvature motion approximated by the univariate median filter: in the lemma, u_{yy} and v_{xx} represent separate mean curvature motion contributions for the u and v channel. Second, there are coupling terms – in the lemma: v_{xy} in the equation for u, and u_{xy} in the equation for v – that promote a joint evolution of the channels. Third, there is an isotropic diffusion term Δu which has no counterpart in the univariate case. Remember that also Figure 3 shows a slight edge-blurring effect of multivariate median filtering.

Proof of Lemma 2. By Taylor expansion of $\boldsymbol{u}$ around $\mathbf{0}$ we obtain within the ϱ-disc D_ϱ around $\mathbf{0}$

$$u \doteq x + ax^2 + by^2 + cxy\,, \qquad (6)$$

$$v \doteq y + dx^2 + ey^2 + fxy\,. \qquad (7)$$

where $\doteq$ denotes equality up to $\mathcal{O}(\varrho^3)$ terms. The inverse function can be written as

$$x \doteq u - au^2 - bv^2 - cuv\,, \qquad (8)$$

$$y \doteq v - du^2 - ev^2 - fuv\,. \qquad (9)$$

Coarse estimates yield that the median of the values $\boldsymbol{u}(x, y)$ for (x, y) in D_ϱ differs from $\mathbf{0}$ by $\mathcal{O}(\varrho^2)$. Let therefore a median candidate point in the (u, v) plane be given as $\boldsymbol{\mu} = (\lambda\varrho^2, \mu\varrho^2)^{\mathrm{T}}$ with $\lambda, \mu = \mathcal{O}(1)$ (i.e., bounded for $\varrho \to 0$). To determine the half-space depth of $\boldsymbol{\mu}$, we consider straight lines through $\boldsymbol{\mu}$ in the (u, v) plane. A parametric representation of such a line $L = L(\varphi)$ is

$$u(t) = \lambda\varrho^2 + tp\,, \qquad v(t) = \mu\varrho^2 + tq \qquad (10)$$

where $p = \cos\varphi$, $q = \sin\varphi$ with the angle φ denoting the direction of the line, and t is a real parameter which also determines an orientation of L.

We are interested in the total weight $w(\varphi)$ of the density of values $\boldsymbol{u}$ within the half-plane on the right side of $L(\varphi)$. The half-space depth of $\boldsymbol{\mu}$ is proportional to the minimum of $w(\varphi)$ for $\varphi \in [0, 2\pi]$.

The line L is mapped to some curve C in the (x, y) plane by the inverse function $\boldsymbol{u} \mapsto \boldsymbol{x}$ from (8), (9). Then, $w(\varphi)$ is proportional to the area of the part of D_ϱ that lies on the right side of C. We will therefore study in the following the part of C within D_ϱ. For sufficiently small ϱ, this is a curve segment corresponding to a parameter interval $[t^-, t^+]$ for t, with $t^\pm = \mathcal{O}(\varrho)$.

We calculate a parametric representation of C by inserting (10) into (8), (9) to obtain

$$x(t) \doteq \lambda\varrho^2 + tp - at^2p^2 - bt^2q^2 - ct^2pq \;, \quad (11)$$
$$y(t) \doteq \mu\varrho^2 + tq - dt^2p^2 - et^2q^2 - ft^2pq \;. \quad (12)$$

By easy estimates, one has $t^\pm \doteq \pm\varrho + r^\pm\varrho^2$ with $r^\pm = \mathcal{O}(1)$. Thus, the intersection points $\boldsymbol{x}^+ = (x(t^+), y(t^+))^\mathrm{T}$, $\boldsymbol{x}^- = (x(t^-), y(t^-))^\mathrm{T}$ of C with the boundary of D_ϱ are given by

$$\boldsymbol{x}^\pm \doteq \pm\varrho \begin{pmatrix} p \\ q \end{pmatrix} + \varrho^2\boldsymbol{\eta}^\pm \;, \quad (13)$$
$$\boldsymbol{\eta}^\pm = \begin{pmatrix} \lambda + r^\pm p - ap^2 - bq^2 - cpq \\ \mu + r^\pm q - dp^2 - eq^2 - fpq \end{pmatrix} \;. \quad (14)$$

As $\boldsymbol{x}^\pm$ are to lie on the boundary of D_ϱ, we have for their Euclidean norms $|\boldsymbol{x}^\pm|$ that $|\boldsymbol{x}^\pm|^2 = \varrho^2$. By $p^2 + q^2 = 1$, this implies $\langle (p, q)^\mathrm{T}, \boldsymbol{\eta}^\pm \rangle = \mathcal{O}(\varrho)$ and thus $r^\pm = r + \mathcal{O}(\varrho)$ and $\boldsymbol{\eta}^\pm = \boldsymbol{\eta} + \mathcal{O}(\varrho)$ with

$$r = ap^3 + (c + d)p^2q + (b + f)pq^2 + eq^3 \;, \quad (15)$$
$$\boldsymbol{\eta} = \begin{pmatrix} \lambda + ap^4 \\ +(c + d)p^3q + (b + f)p^2q^2 + epq^3 \\ -(\lambda p + \mu q)p - ap^2 - bq^2 - cpq \\ \mu + ap^3q \\ +(c + d)p^2q^2 + (b + f)pq^3 + eq^4 \\ -(\lambda p + \mu q)q - dp^2 - eq^2 - fpq \end{pmatrix} \;. \quad (16)$$

From (14) it is evident that the intersection points $\boldsymbol{x}^\pm$ differ from the intersection points $\pm\varrho^2(p, q)^\mathrm{T}$ of the diameter δ_φ of D_ϱ in direction φ with the boundary of D_ϱ just by an offset $\varrho^2\boldsymbol{\eta} + \mathcal{O}(\varrho^3)$. The component of this offset perpendicular to δ_φ is

$$\langle \varrho^2\boldsymbol{\eta}, (-q, p)^\mathrm{T} \rangle = \varrho^2\big(\mu p - \lambda q - dp^3 + (a - f)p^2q + (c - e)pq^2 + bq^3\big) \;. \quad (17)$$

Up to higher order terms $\mathcal{O}(\varrho^3)$, the entire curve C is approximated by a parabola over the diameter δ_φ with height $h(t) = \varrho^2(\mu p - \lambda q) + t^2\big(-dp^3 + (a - f)p^2q + (c - e)pq^2 + bq^3\big)$ for $t \in [t^-, t^+]$. The area on the right of C (i.e., below C) differs from that of the half-disc below the diameter δ_φ by

$$\begin{aligned} \Delta(\varphi) &= \int_{t^-}^{t^+} h(t) + \mathcal{O}(\varrho^3)\,\mathrm{d}t \\ &= 2\varrho^3(\mu p - \lambda q) + \tfrac{4}{3}\varrho^3\big(-dp^3 + (a - f)p^2q \\ &\quad + (c - e)pq^2 + bq^3\big) + \mathcal{O}(\varrho^4) \;. \end{aligned} \quad (18)$$

The half-space depth of $\boldsymbol{\mu}$ is proportional to the minimum of $\pi\varrho^2/2 + \Delta(\varphi)$ for $\varphi \in [0, 2\pi]$.

The sought half-space median is therefore given by those λ, μ for which the minimum of $\Delta(\varphi)$ is largest. It can be proven that the minimum of $\Delta(\varphi)$ differs only by higher-order terms w.r.t. ϱ from that of

$$\begin{aligned} \tilde{\Delta}(\varphi) = &\big(3\mu - \tfrac{3}{4}d + \tfrac{1}{4}(c - e)\big)\cos\varphi \\ &+ \big(-3\lambda + \tfrac{1}{4}(a - f) + \tfrac{3}{4}b\big)\sin\varphi \\ &+ \big(-\tfrac{1}{4}d - \tfrac{1}{4}(c - e)\big)\cos(3\varphi) \\ &+ \big(\tfrac{1}{4}(a - f) - \tfrac{1}{4}b\big)\sin(3\varphi) \end{aligned} \quad (19)$$

where we have inserted $p = \cos\varphi$, $q = \sin\varphi$, and addition theorems. This function is the superposition of a shifted 2π-periodic sine function (combining the $\cos\varphi$, $\sin\varphi$ contributions) and a shifted $2\pi/3$-periodic sine function (combining the $\cos(3\varphi)$, $\sin(3\varphi)$ contributions). Moreover, $\tilde{\Delta}$ is an odd function, such that is maximum and minimum are of equal magnitude and opposite sign. Since only the 2π-periodic part of $\tilde{\Delta}$ depends on λ, μ, it is easy to see that the amplitude of $\tilde{\Delta}$ is minimised (and thus the minimum is maximised) if and only if λ, μ are chosen such that the 2π-periodic contribution vanishes. Again, the neglection of higher order terms in $\Delta(\varphi)$ above entails only a higher-order error in λ, μ. Therefore, the sought median is determined up to higher order terms by

$$\lambda = \tfrac{a}{12} + \tfrac{b}{4} - \tfrac{f}{12} \;, \qquad \mu = \tfrac{d}{4} + \tfrac{e}{12} - \tfrac{c}{12} \quad (20)$$

from which the claim of the lemma follows by virtue of $a = u_{xx}/2$, $b = u_{yy}/2$, $c = u_{xy}$, $d = v_{xx}/2$, $e = v_{yy}/2$, $f = v_{xy}$. □

Proof of Proposition 1. The transfer of the lemma to the general geometric situation of the proposition is analogous to [14, Sect. 3.1.2]. It relies on the observation that for any regular point $\boldsymbol{x} \in \Omega$, transforming the values $\boldsymbol{u}$ in its neighbourhood via the affine transform $\hat{\boldsymbol{u}} = (\mathrm{D}\boldsymbol{u}(\boldsymbol{x}))^{-1}\boldsymbol{u}$ leads to a transformed function $\hat{\boldsymbol{u}}$ with $\mathrm{D}\hat{\boldsymbol{u}} = \mathrm{diag}(1, 1)$ as required by the lemma. Due to the affine equivariance of the

half-space median, the median of $\hat{\boldsymbol{u}}$ yields the median of the original data by the inverse transform. As the PDE system of Lemma 2 is identical to that for the Oja median in [14], the calculations from [14, Eqs. (25)–(26)] for the transform step apply verbatim, and yield the claim of our proposition. □

4. Summary and Outlook

In this work, we have studied the continuous limit of half-space median filtering, one of the possible generalisations of median filtering of grey-value images to multi-channel images, in the bivariate case. We have proven a result already conjectured in [15] stating the approximation of a particular PDE by this filter. The result is embedded in the context of previous work on PDE approximation by multivariate median filters, see [14], and is a step on the way to a deeper understanding of multivariate median filters for signals and images.

An interesting fact is that despite clear differences in the practical outcome of the corresponding filters on discrete images (see Figure 1), the affine equivariant Oja median and half-space median filter approximate the same PDE. This indicates that they can be seen as different discrete realisations of *one* underlying fundamental multivariate median filter, despite the substantial differences in their underlying discrete concepts (see the discussion in [15]).

As mentioned earlier, the focus of our work was in the theoretical domain. Further study of the practical applicability of half-space median filtering is a subject of ongoing work. In particular, algorithmic efficiency issues will require further investigation. Moreover, bivariate images as considered here are a rare exception in practice (with two-dimensional optic flow fields being the most relevant case, see [14]). A much greater role is played by images with three (such as RGB colour images or tensor fields in two dimensions) or even more channels (multispectral images, tensor fields in three dimensions). Extension of the theoretical investigation to three and more channels is therefore another important goal for future research.

References

[1] J. Astola, P. Haavisto, and Y. Neuvo. Vector median filters. *Proceedings of the IEEE*, 78(4):678–689, 1990.

[2] T. L. Austin. An approximation to the point of minimum aggregate distance. *Metron*, 19:10–21, 1959.

[3] V. Barnett. The ordering of multivariate data. *Journal of the Royal Statistical Society A*, 139(3):318–355, 1976.

[4] C. Gini and L. Galvani. Di talune estensioni dei concetti di media ai caratteri qualitativi. *Metron*, 8:3–209, 1929.

[5] F. Guichard and J.-M. Morel. Partial differential equations and image iterative filtering. In I. S. Duff and G. A. Watson, editors, *The State of the Art in Numerical Analysis*, number 63 in IMA Conference Series (New Series), pages 525–562. Clarendon Press, Oxford, 1997.

[6] R. Y. Liu. On a notion of data depth based on random simplices. *The Annals of Statistics*, 18(1):405–414, 1990.

[7] H. Oja. Descriptive statistics for multivariate distributions. *Statistics and Probability Letters*, 1:327–332, 1983.

[8] A. H. Seheult, P. J. Diggle, and D. A. Evans. Discussion of paper by V. Barnett. *Journal of the Royal Statistical Society A*, 139(3):351–352, 1976.

[9] C. Spence and C. Fancourt. An iterative method for vector median filtering. In *Proc. 2007 IEEE International Conference on Image Processing*, volume 5, pages 265–268, 2007.

[10] J. W. Tukey. *Exploratory Data Analysis*. Addison–Wesley, Menlo Park, 1971.

[11] J. W. Tukey. Mathematics and the picturing of data. In *Proc. of the International Congress of Mathematics 1974*, pages 523–532, Vancouver, Canada, 1975.

[12] A. Weber. *Über den Standort der Industrien*. Mohr, Tübingen, 1909.

[13] E. Weiszfeld. Sur le point pour lequel la somme des distances de n points donnés est minimum. *Tôhoku Mathematics Journal*, 43:355–386, 1937.

[14] M. Welk. Multivariate median filters and partial differential equations. *Journal of Mathematical Imaging and Vision*, 56:320–351, 2016.

[15] M. Welk. Multivariate medians for image and shape analysis. Technical Report 1911.00143 [eess.IV], arXiv.org, 2019.

[16] M. Welk and M. Breuß. The convex-hull-stripping median approximates affine curvature motion. In M. Burger, J. Lellmann, and J. Modersitzki, editors, *Scale Space and Variational Methods in Computer Vision*, volume 11603 of *Lecture Notes in Computer Science*, pages 199–210. Springer, Cham, 2019.

[17] M. Welk, C. Feddern, B. Burgeth, and J. Weickert. Median filtering of tensor-valued images. In B. Michaelis and G. Krell, editors, *Pattern Recognition*, volume 2781 of *Lecture Notes in Computer Science*, pages 17–24. Springer, Berlin, 2003.

360° Monitoring for Robots Using Time-of-Flight Sensors

Thomas Maier, Birgit Hasenberger
BECOM Systems GmbH
{thomas.maier,birgit.hasenberger}@becom-group.com

Abstract. *In this paper, we present a system based on multiple Time-of-Flight (ToF) 3D sensors paired with a central processing hub for integration into robots or mobile machines. This system can produce a 360° view from the robot's perspective and enables tasks ranging from navigation and obstacle avoidance to human-robot collaboration.*

1. Introduction

Today's e-commerce growth and the paradigms of Industry 4.0 in the manufacturing space present new challenges for robotic systems [1]. In order to increase mobility and autonomy of such systems they need to gather and interpret as much information as possible from their surroundings. Approaches for collision avoidance with 1D time-of-flight sensors have been explored [2], the use of several high-resolution sensors would enable applications like human pose estimation and gesture recognition as well as automation tasks such as handling of goods. Close collaboration and additional functionalities are made possible with the setup proposed in this paper which consists of intrinsic 3D Time-of-Flight sensors that can cover 360° around a robot's arm or chassis.

2. System architecture

The multi-ToF platform[1] consists of a central processing module based on a NVIDIA Tegra TX2 processor (the hub) and multiple camera modules that work in parallel (the frontends). This platform architecture allows for the integration of various sensor and camera types.

The ToF sensor frontend for the platform is a compact module designed for close-range detection. Table 1 summarises the technical data and performance of two frontend variants. The frontend is connected

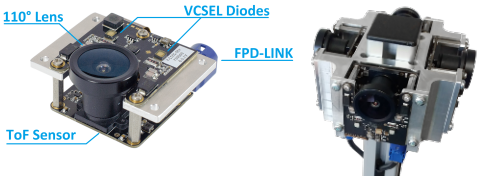


Figure 1. Image of a front-end including descriptions of its components *(left)* and a ring of four front-ends *(right)*.

	QVGA frontend	VGA frontend
Resolution	304 px x 240 px	640 px x 480 px
Field of view	110° x 82°	110° x 82°
Distance range	0.1 – 1.5 m	0.1 – 2.0 m
Operating wavelength	850 nm	940 nm
Framerate	40 Hz	30 Hz

Table 1. Frontend specifications.

to the hub via FPD-Link III, using a cable that also provides the power supply. Four frontends can be arranged as a ring to allow a 360° coverage.

The hub controls the frontends, performs calibration and correction operations on the incoming data, and ultimately calculates depth maps or point clouds. The data is transmitted from the hub via a Gigabit Ethernet connection and supports ROS to gather the individual data streams. The following operations are performed on the hub:

1. Synchronization and triggering
2. Acquisition and depth map calculation
3. Corrections (temperature, FPPN, distance offset, intrinsic and extrinsic)
4. Filtering (spatial and temporal)
5. 3D point cloud calculation
6. Registration and transformation

[1] https://www.becom-group.com/goto/multi-tof-platform

Operations 1 and 6 are specific to multi-camera systems and are therefore described in more detail in the following sections. The remaining CPU/GPU performance on the hub is available for AI and deep-learning applications.

3. Synchronization

Imaging systems that rely on multiple active sensors inevitably require a synchronization. In the case of the multi-ToF platform, synchronization serves two purposes: On the one hand, it avoids interference effects between the sensors, and on the other hand, it simplifies the registration of the point clouds produced by the individual frontends. The hub can synchronize multiple frontends by using a hardware trigger to start the acquisition of individual frontends. This could be done in a round-robin scheme or by triggering opposite sensors at the same time to avoid interference.

4. Point cloud registration

Each ToF-sensor frontend produces a 2D depth map which can be converted into a 3D point cloud. A consistent 360° view of the environment necessitates the registration of these individual point clouds in a common world coordinate system. Through an extrinsic calibration all sensors of the ring can be combined in a single point cloud which can be transformed in a robot or world coordinate system given a known position of the robot's joints.

5. Advantages and performance

With today's ToF technology, cameras are capable of detecting objects with high framerates and low latencies. Active lighting ensures that data quality is independent from ambient conditions to a high degree. The exact distance measurement accuracy is dependent on the target's reflectivity and distance, but the user can expect a relative accuracy of 1 % based on the distance.

Considering the system's performance in the context of machine learning, ToF cameras provide useful additional information compared to, for example, 2D RGB cameras: Objects can be more easily spatially separated using the 3D point cloud and the corresponding IR greyscale image can be employed when training a network. Training labels can easily be transferred between the four ToF channels (X, Y, Z, and amplitude) at pixel precision. As a result, ToF cameras reveal more information about the observed scene, but labelling the data does not require additional effort. The recognition performance of deep learning algorithms in particular benefits from an increase in the amount of available data.

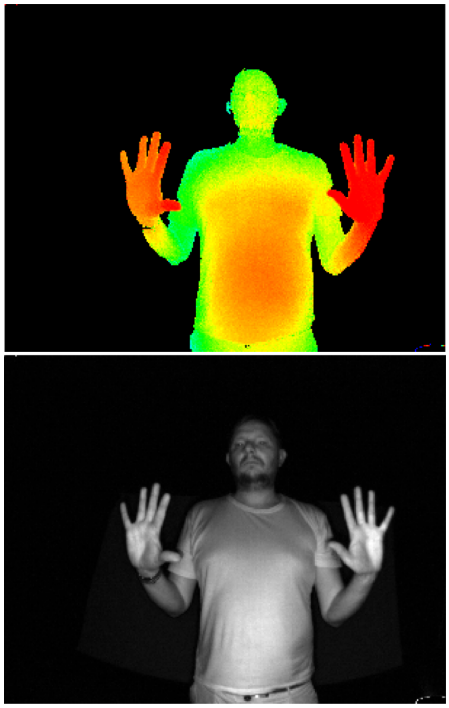


Figure 2. Depth image *(left)*, with red indicating smaller and green larger distances, and the corresponding IR greyscale image *(right)*.

6. Conclusion

In this paper we have presented a hardware platform which uses multiple ToF Sensors and a central processing hub to generate a high-resolution point cloud around an autonomous machine which enables collaborative and safety functions. Further work will include a synchronization of multiple machines working in close proximity using a clock synchronization mechanism over state of the art wireless connectivity hardware.

References

[1] U. Behrje, M. Himstedt, and E. Maehle. An autonomous forklift with 3d time-of-flight camera-based localization and navigation. In *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, pages 1739–1746. IEEE, 2018.

[2] S. Kumar, S. Arora, and F. Sahin. Speed and separation monitoring using on-robot time-of-flight laser-ranging sensor arrays. In *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*, pages 1684–1691. IEEE, 2019.

Towards Identification of Incorrectly Segmented OCT Scans

Verena Renner and Jiří Hladůvka
Pattern Recognition and Image Processing Group
TU Wien
e1527272@student.tuwien.ac.at jiri@prip.tuwien.ac.at

Abstract. *Precise thickness measurements of retinal layers are crucial to decide whether the subject requires subsequent treatment. As optical coherence tomography (OCT) is becoming a standard imaging method in hospitals, the amount of retinal scans increases rapidly, automated segmentation algorithms are getting deployed, and methods to assess their performance are in demand.*

In this work we propose a semi-supervised framework to detect incorrectly segmented OCT retina scans: ground-truth segmentations are (1) embedded in 2D feature space and (2) used to train an outlier scoring function and the corresponding decision boundary.

We evaluate a selection of five outlier detection methods and find the results to be a promising starting point to address the given problem. While this work and results are centred around one concrete segmentation algorithm we sketch the possibilities of how the framework can be generalized for more recent or more precise segmentation methods.

1. Introduction

It is known that frequent eye screening helps to early-diagnose the diabetic macular edema (DME) [14] and therefore raises the effectiveness of needed treatments. Additionally, the number of age-related macular degeneration (AMD) patients is increasing, because of ageing population [9], as well as those suffering from DME due to the rising number of diabetes cases. OCT technology is nowadays minimally invasive, very fast, and therefore widely spread, so that a large number of OCT scans needs to be preprocessed automatically. Ophthalmological departments are developing or deploying systems to deal with the large amount of OCT data produced. One such instance to segment retinal layers from OCT scans is based on the work [5]. While accurate in most of cases, the method occasionally exhibits imperfections. An improvement is desirable, as the correct segmentation is essential for further automatic evaluation of OCT scans. This is because the thickness of the retinal layers is highly related to the presence of diseases, like AMD or DME [5]. They are caused by intraretinal and subretinal fluids, leading to a swelling of the retinal layers [10], exerting pressure on the light-receptors damaging them and thus eyesight.

Imperfections in segmentation can be caused by different reasons such as bad contrast of parts of the scan, noise, artefacts or an unsupported edge-case of the segmentation algorithm.

This work aims to support the identification of incorrectly segmented OCT scans with a two-fold purpose in mind. First, it is of interest to increase the trust of ophthalmologists in the algorithm by flagging segmentations that may potentially require manual inspection. Second, to improve segmentation algorithms, it is desirable to automatically identify incorrect segmentations of previously unseen scans and focus on improvements for such cases.

2. Dataset

A set of 100 OCT scans, each accompanied with both manual ground truth (GT) and algorithmic (A) segmentation [5] have been provided for this study. Each OCT scan is a stack of 200 1024×200 gray scale images. Both the ground truth and the algorithmic segmentation are available as slice-wise boundaries of 13 retina layers. Figure 1 shows boundary examples of the first retina layer (L1). There is no expert assessment available on whether the algorithmic segmentations are accepted as correct or not.

For legal issues, this dataset is currently unavailable for public use.

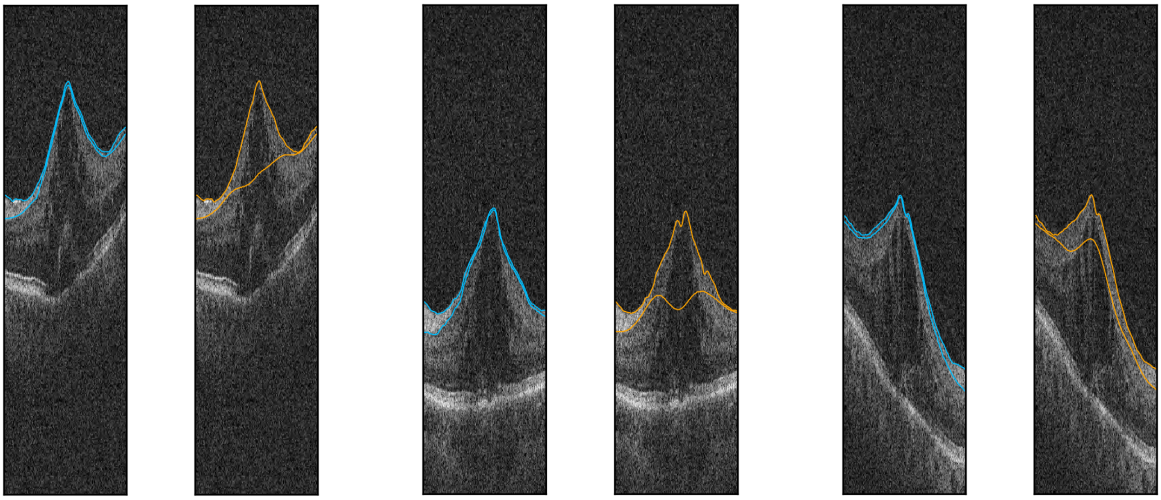


Figure 1: Three examples (scans 1, 2, and 5) of ground truth (left) and incorrect algorithmic (right) segmentations of the first retina layer (L1) in mid-stack slices.

3. Method

In order to identify incorrectly segmented OCT scans we suggest in section 3.1 to embed the segmentation results in as few dimensions as possible. While this is certainly motivated by curse of dimensionality it is additionally motivated by an increase of interpretability – ophthalmologists may desire to visually relate a particular case to cases inspected previously.

Methods of outlier detection can be divided in three branches [6]. Supervised classification, when both inliers and outliers are labeled and in balance; unsupervised when training data of both inliers and outliers are unlabeled; and semi-supervised when training data consists only of observations describing normal behavior. In section 3.2 we follow semi-supervised methods for the following reasons. First, there is no assessment of algorithmic segmentations available and we only can roughly estimate the class based on some metric (e.g., the Dice coefficient). Second, the outlier class (wrong segmentations) is expected to be under-represented. Third, it is likely there are several sources of segmentation error which could map to low-density clusters. We aim to detect outliers in low-density regions, too. We model the distribution of the inliers (correct segmentation) and compare the test points to this distribution.

3.1. Area curves and their representation

While for each retinal layer a list of region properties can be thought of, for sake of interpretability the slice-wise area values are of special interest. Furthermore, the focus of this work was restricted to layer 1. This decision is based on the observation that a segmentation error in L1 layer propagates to subsequent layers while correct L1 segmentations tend to correlate with correctly segmented scans.

For each segmented OCT, we introduce the vector $\mathbf{a} = [a_0, \ldots, a_{199}]^\top$ of layer-1 area values and refer to is as the area curve. Examples of how area curves look like for both ground truth and algorithmic segmentations are given in figure 2.

Looking at the (orange) area curves calculated from the algorithmic segmentation, which are of the main interest, two types of shape appear: Those exhibiting a maximum (cf., scan 1, 2 or 5 of figure 2), or a minimum (cf. scan 0) around the middle of the slices.

In healthy eyes, the layers get thinner around the cavity of the fovea [11], causing the area curves to exhibit a global minimum and tend to be convex. The first hypothesis about the curves with dominant concave bumps therefore was that they may correspond to pathologies where fluid intruded into the retinal layers and caused them to thicken.

Closer inspection of the corresponding scans and a comparison to the (blue) GT area curves, however, quickly disproved this hypothesis and revealed that the concave bumps tend to correspond to failures in segmentation. Further investigations revealed that the issue of a too thick segmented layer 1 appeared in all scans that exhibit a global maximum in the area curve or tend do be concave.

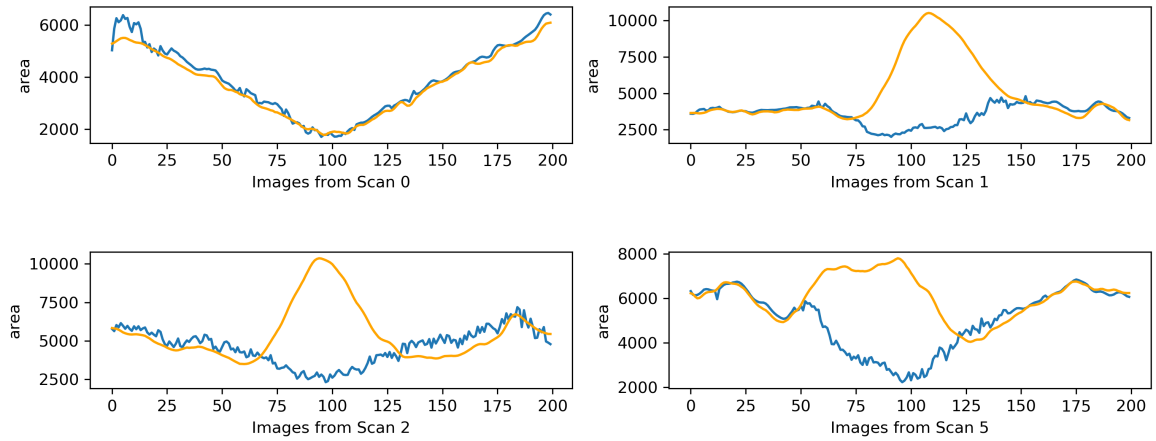


Figure 2: Examples of area curves resulting from ground truth (blue) and algorithmic (orange) segmentations. Scan 0 is an example of correct segmentation, the remaining three cases (scans 1, 2, 5) correspond to incorrect segmentations shown in figure 1.

3.1.1 Curve Embedding

To grasp the convex-vs-concave nature of the area curves and to embed them in a lower dimensional space we chose to approximate them by second order polynomials $a(x; \mathbf{w}) \approx w_0 + w_1 x + w_2 x^2$ and to represent them by the three regression coefficients w_i.

Following [3] the regression coefficients $\mathbf{w} = [w_0, w_1, w_2]^\top$ for each area curve are calculated by means of regularized least-squares, i.e, by solving $\mathbf{w} = (\lambda \mathbf{I} + \mathbf{\Phi}^\top \mathbf{\Phi})^{-1} \mathbf{\Phi}^\top \mathbf{a}$, where λ is the regularization term, $\mathbf{I}$ the 3×3 identity matrix, $\mathbf{\Phi}$ the 200×3 design matrix with rows $[1, x, x^2]$, and x indexes the slices $x \in \{0..199\}$.

The optimal regularization coefficient was determined close to zero $\lambda \approx 0$, which can be explained by the fact that fitting a low-grade polynomial to 200 values does not suffer from overfitting. This reduces the curve fitting to ordinary least squares, i.e., multiplication of the area curve vector by the psuedoinverse of the design matrix: $\mathbf{w} = (\mathbf{\Phi}^\top \mathbf{\Phi})^{-1} \mathbf{\Phi}^\top \mathbf{a} = \mathbf{\Phi}^\dagger \mathbf{a}$.

3.1.2 Regression Coefficients

Regression coefficients corresponding to all 100 ground truth (blue) as well as algorithmic (orange) segmentations are scatter-plotted in the first row of figure 3. Its second row shows the three corresponding kernel density estimation (KDE) plots.

The two w_0 KDE plots indicate very similar distributions and therefore the w_0 coefficients do not seem to be discriminative.

The too-thick segmented layers are mapped to concave area curves. Therefore, the distribution of w_2 coefficients is of special interest, as they are responsible for the positive/negative curvature of the polynomials. Looking at the KDE plot of w_2 coefficients, there is a high peek from the ground truth coefficients between 0 and 0.5, showing that there are almost no negative w_2 coefficients. Therefore the assumption that ground truth curves tend to exhibit convexity (positive curvature) holds. In contrast, the orange KDE resulting from algorithm segmentations is more flat in the GT area and also exhibits a minor peek around -0.5. This indicates the presence of a cluster of negative w_2 values, which corresponds to concave area curves. This distribution can be confirmed looking at scatter plots including w_2. For example in the w_1–w_2 plot there is a (blue) cluster formed by ground truth coefficients while several negative w_2 algorithm coefficients are scattered outside of it.

Interestingly the w_1 coefficients exhibit a very similar behaviour to the w_2 coefficients: almost no positive w_1 GT coefficients and a tendency to bimodal distribution of the algorithm ones forming a small peek around value of 100.

The highly correlated coefficients w_1 and w_2 encourage for further dimensionality reduction. Indeed,

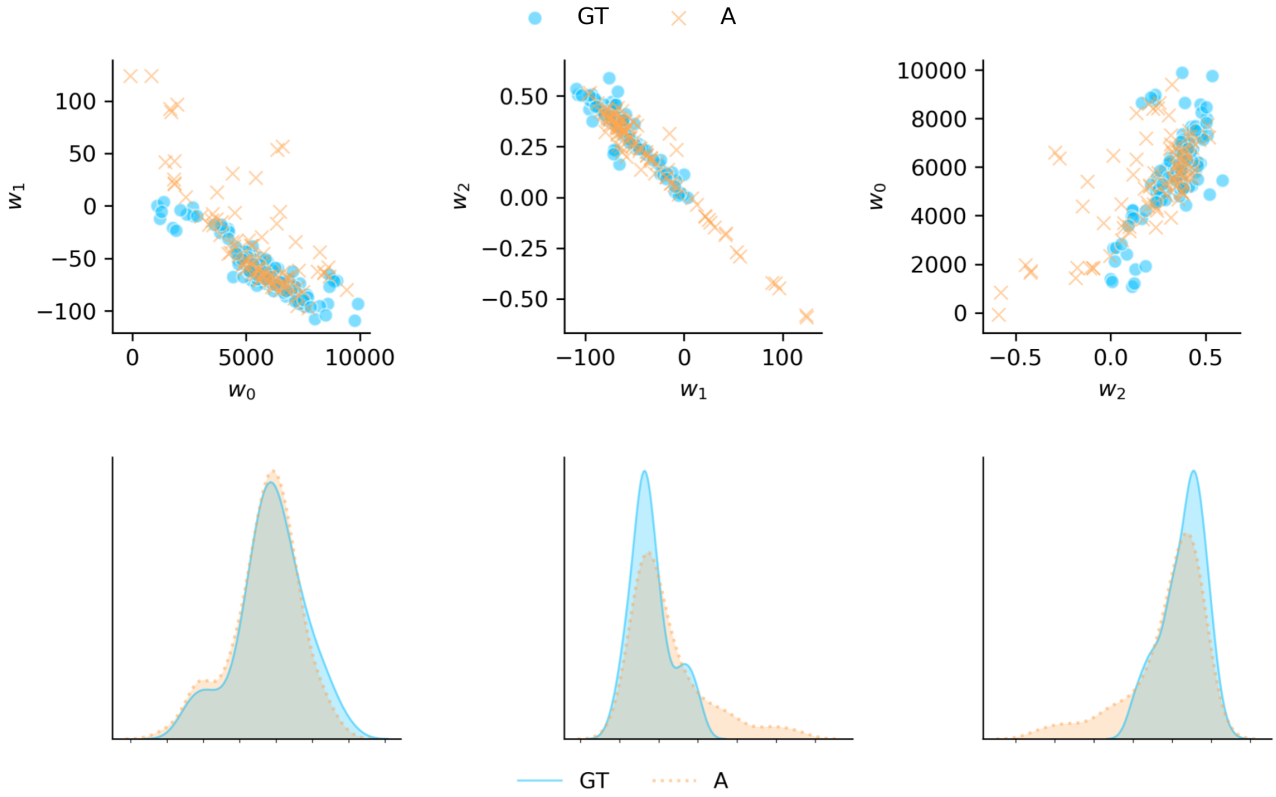


Figure 3: Scatter and kernel density estimation plots of the L1-area curve regression coefficients. The blue and orange dots/curves correspond to the respective coefficients of ground truth and algorithmic segmentation.

an interactive 3D scatter plot revealed the points close to a 2D linear manifold embedded in three dimensions. Projection of both ground-truth and algorithm-segmentation coefficients onto the first two PCA eigenvectors yields 2D scatter plot shown in the left part of figure 4.

In the following the problem of identifying incorrect segmentations is thus cast to outlier detection in a 2D feature space.

3.2. Outlier Detection Using Projected Regression Coefficients

Our approach to outlier detection is a semi-supervised one: we reuse the ground-truth coefficients to fit a model that represents the expected segmentation behavior. Subsequently the likelihood of an algorithmic segmentation to be generated by the learned model is tested.

While there is a broad spectrum of methods for outlier (novelty) detection, we show a digest of 5 algorithms resulting from our experiments and discuss their performance.

Feature Bagging (FB) [7] fits several base detectors on sub-samples of the dataset and use averaging to combat over-fitting. We used the LOF (see below) as the base detector.

Nearest Neighbors (KNN) [2] the distance of the sample to its most distant k-th neighbor is used as the outlier score. We set $k = 5$.

Local Outlier Factor (LOF) [4] Samples with much lower local density than their neighbors are declared as the outliers. The local density was estimated by 20 nearest neighbors.

Minimum Covariance Determinant (MCD) [12] fits the minimum covariance determinant model to the data. The outlier-ness of a sample is proportional to its Mahalanobis distance.

One-class SVM (OCSVM) introduced in [13] aims to find a smooth boundary modelling a user-specified probability that randomly drawn point will land outside.

4. Results and Discussion

To evaluate the outlier detectors quantitatively, notion of positives (incorrect segmentation) and negatives is necessary for the test data, i.e. for algorithmic

segmentations. As this information was not present we chose to disambiguate the two classes by setting a Dice coefficient threshold. To figure out a sufficiently high Dice threshold we refer to the score-vs-dice scatter plot in the middle column of figure 4. Here the blue margin corresponds to the ground-truth region, proposed by the detector. The orange cluster within this margin then corresponds to true negatives (correct segmentations), and suggests the dice threshold of 0.87.

Figure 4 shows three plots for each of the five methods. In the following its columns are described in detail.

Left column: in addition to feature scatter plot the decision boundary and the scoring function of the respective detector are shown. In the following texts the orange test points falling outside the blue region will be referred to as the positives, points inside the blue region as the negatives.

Middle column shows scatter plots of Dice vs outlier scores. The horizontal line is the Dice threshold. The vertical lines are the thresholds of the scoring function proposed by the respective algorithms. The four quadrants correspond to TNs, FPs, FNs, and TPs, respectively. These four numbers are typeset in the top center of the plot and the recalls and precisions computed thereof are displayed in the titles.

Right column shows the ROC and Precision-Recall curves corresponding to the possible thresholds in the scoring function. The areas under these curves are abbreviated by auROC and auPR, respectively, and are displayed in the title.

The performance numbers are summarized in Table 1. In terms of precision, the areas under ROC and PR, the kNN seems to be the method of choice. However, the OCSVM wins in term of recall, because of its steep narrow margin which determines the outlier score. While LOF and FB are of lower recall, they are less over-fitted than earlier two, and we can observe an improvement when an ensemble of LOFs is aggregated into the FB. The MCD is easily interpretable but unfortunately not performing well.

Looking at the result of the well-fitting OCSVM, there are four FNs with a low Dice coefficient. Investigation on these revealed that such cases indeed might appear, because the area curves of the ground truth do not show a minimum around the middle of the slices, but have a nearly a rising shape. While the segmentation algorithm did not perform well on these scans, it still exhibits a convex fit to the area curve.

method	Rec.	Prec.	auROC	auPR
FB	0.73	0.95	0.96	0.93
KNN	0.77	1.00	0.97	0.94
LOF	0.65	0.89	0.96	0.91
MCD	0.54	0.88	0.95	0.87
OCSVM	0.85	0.92	0.88	0.89

Table 1: Summary of results

Analyzing the two false positives, one of them appeared close to the OCSVM boundary. The less over-fitted detectors (e.g the MCD), however, have classified this point correctly. The second false positive was a FP in all methods, except for the KNN. This could be because the ground truth data again shows an unusual shape: in contrast to the other ground truth shapes it starts with a high maximum, then falls down, but does not rise up again. There are few additional ground truth curves having this kind of shape which we consider unusual. When the segmentation algorithm yields such a shape, it is more likely to be a wrong segmentation.

Whether an ROC curve should be used to assess an outlier detector depends on the imbalance of the test set. In the current setting, the segmentation algorithm [5] does not seem to be mature enough as it produces around 25 percent of incorrect segmentation. As more reliable segmentation methods will be developed, the test set becomes increasingly more imbalanced and the validation by ROC and its area will have to be replaced by the precision-recall curves.

5. Conclusion and Future Work

We proposed a semi-supervised method to detect incorrectly segmented OCT retina scans: ground-truth segmentations are used, after feature extraction and projection to 2D, to train the decision boundary and the outlier scoring function. This function is subsequently used to flag the incorrectly segmented scans.

We evaluated a selection of five outlier detection methods and find the results to be a promising starting point to address the given problem.

While in this work the data-pipeline components are tailored to a specific segmentation algorithm and its pitfalls, we would like to sketch how the presented approach can be generalized. Firstly, higher-degree polynomials (i.e., more regression coefficients) could be used if it turns out that the segmentations can not

be discriminated by the concave/convex shapes. Secondly, we concentrated only on description of layer 1, as imperfections in its segmentation propagated to subsequent layers. As the segmentation algorithms mature, descriptors of remaining layers could be incorporated. With an increased number of features, the ensemble-based detectors (FB in this work) may improve in their performance. Finally, after the segmentation algorithms become very advanced, it may turn out that the area-related descriptors loose their discriminative power and a need for completely new set descriptors may arise. In the proposed semi-supervised framework, the manually crafted features can be replaced by ones proposed by auto-encoders [1] or generative adversarial neural networks [8].

References

[1] C. C. Aggarwal. Outlier analysis. In *Data mining*, pages 75–79. Springer, 2015.

[2] F. Angiulli and C. Pizzuti. Fast outlier detection in high dimensional spaces. In *European Conference on Principles of Data Mining and Knowledge Discovery*, pages 15–27. Springer, 2002.

[3] C. M. Bishop. *Pattern Recognition and Machine Learning*, chapter 3.1.4: Regularized least squares, pages 144–145. Springer, 2006.

[4] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander. LOF: Identifying density-based local outliers. In *ACM sigmod record*, volume 29, pages 93–104. ACM, 2000.

[5] M. K. Garvin, M. D. Abramoff, X. Wu, S. R. Russell, T. L. Burns, and M. Sonka. Automated 3-d intraretinal layer segmentation of macular spectral-domain optical coherence tomography images. *IEEE Transactions on Medical Imaging*, 28(9):1436–1447, Sep. 2009.

[6] V. Hodge. A survey of outlier detection methodologies. *Artificial Intelligence Review*, 22:85–126, 10 2004.

[7] A. Lazarevic and V. Kumar. Feature bagging for outlier detection. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 157–166. ACM, 2005.

[8] Y. Liu, Z. Li, C. Zhou, Y. Jiang, J. Sun, M. Wang, and X. He. Generative adversarial active learning for unsupervised outlier detection. *IEEE Transactions on Knowledge and Data Engineering*, 2019.

[9] P. J. Mekjavic, V. J. Balciüniene, L. Ceklic, J. Ernest, Z. Jamrichova, Z. Z. Nagy, I. Petkova, S. Teper, I. G. Topcic, and M. Veith. The burden of macular diseases in central and eastern Europe —- implications for healthcare systems. *Value in Health Regional Issues*, 19:1–6, 2019.

[10] T. Otani, S. Kishi, and Y. Maruyama. Patterns of diabetic macular edema with optical coherence tomography. *American Journal of Ophthalmology*, 127(6):688–693, 1999.

[11] Pro Visu Foundation. Fovea Centralis. https://www.provisu.ch/cgi/en/anatomical-structure.pl?en+alp+F+A09.371.729.522.436, 2018. [Online; accessed 13-October-2019].

[12] P. J. Rousseeuw and K. V. Driessen. A fast algorithm for the minimum covariance determinant estimator. *Technometrics*, 41(3):212–223, 1999.

[13] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson. Estimating the support of a high-dimensional distribution. *Neural computation*, 13(7):1443–1471, 2001.

[14] W. D. Strain, X. Cos, and C. Prünte. Considerations for management of patients with diabetic macular edema: Optimizing treatment outcomes and minimizing safety concerns through interdisciplinary collaboration. *Diabetes Research and Clinical Practice*, 126:1–9, 2017.

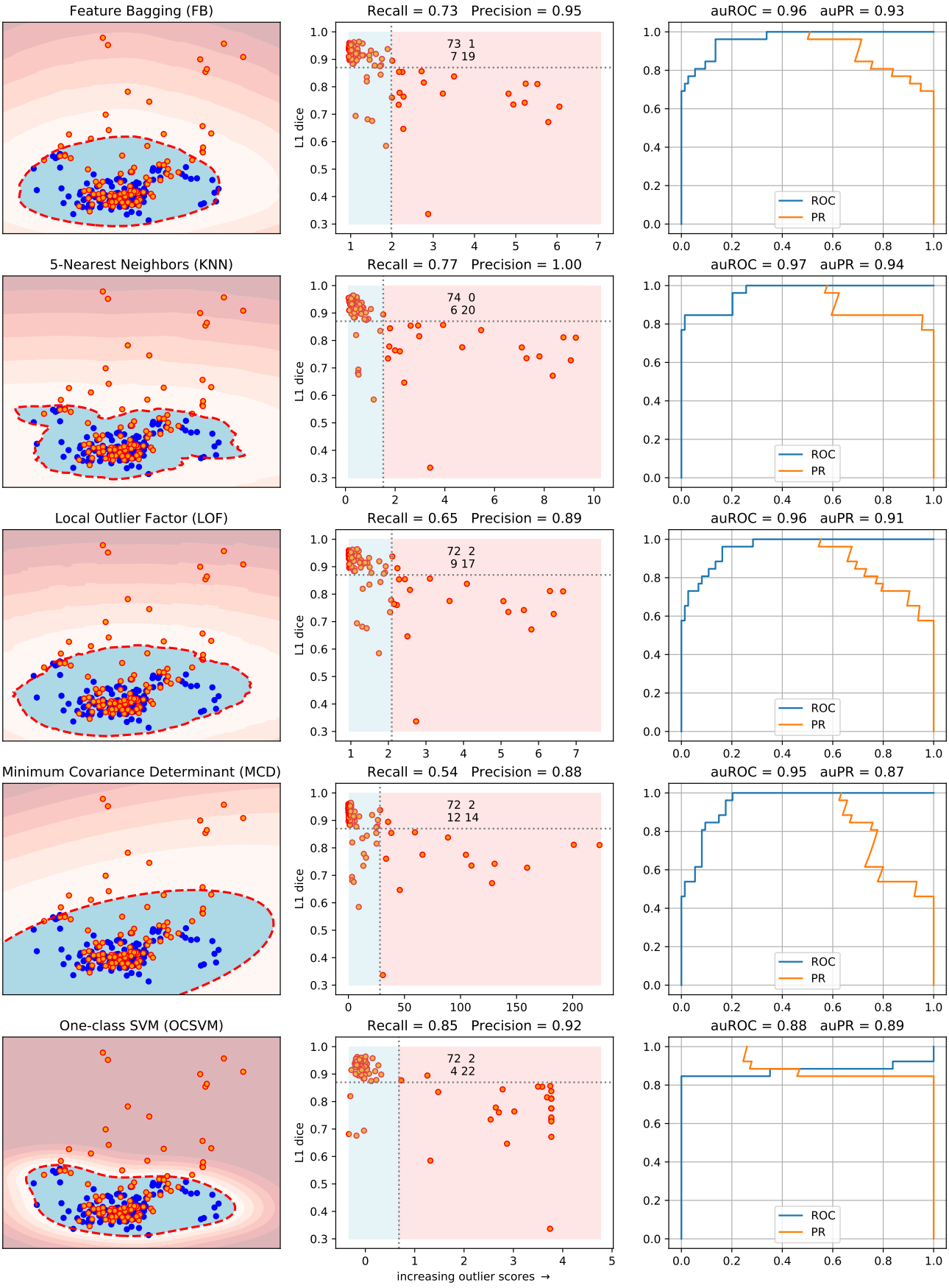


Figure 4: Selected outlier detectors and their performance on test set, i.e., the segmentation results of the algorithm.

Evaluating Counter Measures against SIFT Keypoint Forensics

Muhammad Salman, Andreas Uhl
Department of Computer Sciences, University of Salzburg
uhl@cs.sbg.ac.at

Abstract. *Forensic analysis is used to detect image forgeries e.g. the copy move forgery and the object removal forgery. Counter forensic techniques (methods to fool the forensic analyst by concealing traces of manipulation) have become popular in the game of cat and mouse between the analyst and the attacker. Methods to counter forensic techniques based on SIFT keypoints are being analysed in this paper (aka anti-forensic techniques), with particular emphasis on keypoint removal in the context of copy move forgery detection. Local smoothing is suggested in this paper and turns out to be a highly attractive alternative to techniques investigated in literature so far.*

1. Introduction

In the past, images were considered as an authentic source of information – with increasing popularity and the availability of low-cost image editing software such as Adobe photoshop, corel paint shop and GIMP the truthfulness of an image can no longer be taken for granted. Among other forgery types, copy move forgery and object removal forgery are the most prominent ones. In a *copy move forgery*, a part of the image itself is copied and pasted into another part of the same image to conceal an important object or information, or to conceal that an object has been removed from the image in an *object removal forgery*. In most cases of image forgery, it is extremely difficult to distinguish between an original image and the forged one. Therefore, it is required to develop methods/techniques to assess the authenticity of an image – Digital Image Forensics (DIF [19]) has served this purpose to a large extent. Whenever an image is forged, there are some traces which are left behind in the forged image. These traces are useful for the forensic researcher to detect a forgery.

A wide range of DIF forgery detection techniques have been established in the recent years [4, 6, 21]. Besides recent deep learning based schemes, techniques relying on Scale Invariance Feature Transform (SIFT) keypoints have been shown to be effective. In particular, SIFT keypoints [12] have been proposed to reveal copy move forgeries [6] and image cloning [17], as well as to detect copyrighted material using CBIR techniques [9].

Attackers are making it difficult to apply these techniques by developing counter forensic techniques, i.e. by minimising those traces left behind in forged images. In the context of SIFT keypoint forensics, this is done by manipulating SIFT keypoints, e.g. removing existing ones or injecting fake key points to fool the forensic techniques. This paper is a contribution to such counter forensic approaches against SIFT-keypoint forensic techniques. In particular, we focus on SIFT keypoint removal techniques. Section 2 reviews corresponding techniques as proposed in literature and suggest a new approach. Section 3 is devoted to an extensive empirical evaluation, looking at the tradeoff among image quality, keypoint removal effectiveness as well as the generation of new keypoints. In the conclusion we discuss results obtained and give an outlook to further work in this direction.

2. SIFT Keypoint Removal Techniques

The simplest approach, *global smoothing* (GS), reduces the potential keypoints at the level of difference of Gaussian (DoG) by Gaussian smoothing (which flattens the pixel values of an image), e.g. [1] applies a Gaussian filter with $\sigma = 0.7$ and window size 3×3 as a good compromise between amount of deleted keypoints and overall visual quality of an image. A more sophisticated approach is to first apply GS (the original paper [9] suggests to employ $\sigma = 1.3$), detect remaining keypoints, and apply *local smoothing* (LS) in patches around detected key-

points, with size 3×3 to 7×7 pixels (denoted as GS+LS).

Another strategy to remove SIFT keypoints is the *collage attack* (CA) [10], which substitutes an original image patch (patch containing a keypoint) with another patch (containing no keypoint) of the same size contained in a pre-computed patch dictionary. The new patch must not contain SIFT keypoints and should be as similar as possible to the original one according to some similarity criteria (e.g., [1] created a dictionary of about 120,000 patches and chose histogram intersection distance, widely used in image retrieval applications [22], as a patch similarity measure. The same approach is used in experiments of [3].

Removal with minimum distortion (RMD) [9] adaptively calculates a small image patch and adds it to the neighbourhood of the key point such that the overall operation results in a minimum least-square distortion in the keypoint neighbourhood under the condition that the keypoint is removed. Finally, the *classification based attack* (CLBA) presented by [1] arranges GS+LS, CA, and RMD into an iterative procedure which first detects SIFT keypoints, classifies them into distinct classes, and subsequently applies one of the three individual removal techniques to the suited classes.

For all these techniques, [2] suggested to remove only one of the matching keypoints from each matching keypoints pair in case of preventing to detect copy move forgeries. There are also forensic techniques to counter those anti-forensic keypoint removal methods (see e.g. [7, 16].

As GS has significant impact on image quality (as we shall see as well in the next section), also the combination with LS (i.e. GS+LS) is affected by this quality impact. Therefore, we introduce a new technique to remove SIFT keypoints called *local smoothing* (LS), and compare the various performance indicators to already existing (smoothing) techniques i.e. GS, GS+LS, and CA.

3. Experiments

3.1. Experimental Settings

With respect to software and tools, we mainly used Matlab 2014a [14] (on Windows 7 64bit) with some internal toolboxes (parallel toolbox for fast computation, image processing toolbox) and the external library *vl_feat* [20] (the latter to smooth images and to compute SIFT keypoints; we have chosen *Edge Thresh = 12* to control the number of keypoints used). For the computation of image quality metrics (IQM) (*PSNR, VSNR, UQI, SSIM*), we used the MatTrix MuX visual quality assessment package [15]. As experimental data, we used the first 100 images (i.e. from *ucid00001.tif* to *ucid000100.tif*) from the Uncompressed Image Database (UCID) [18] for experiments for keypoint removal methods assessment. For the CA, we created a keypoint-free patch dictionary from all images using overlapping patches.

For experiments with respect to detecting actual copy move attacks, we combined two datasets to result in 100 images (50 actually forged images and 50 original images). Forged images are taken from a public dataset for assessing forensic techniques [5] (see Fig. 5 for examples), which contain simple translated copies of objects/regions, while the "original" images are taken from the RAISE dataset [8] from the `BUILDING PHOTO` category (see Fig. 6). The latter data has been included to determine the methods' robustness against indicating false positives[1]. In keypoint removal for countering copy move detection, we removed only one keypoint from each matching pair of keypoints as suggested.

3.2. Experimental Results

In order to assess the quality of the image after removing keypoints, we used different IQM, i.e. `PSNR, SSIM, VSNR, and UQI`. Fig. 1 compares three different techniques, i.e. GS, GS+LS, and LS. In GS+LS, an image is smoothed first globally with $\sigma = 1.3$ as suggested in literature and afterwards patches containing keypoints (of different sizes) are smoothed locally. In the plots, different smoothing strength (different σ values) is depicted on the **X axis**, while the **Y axis** represents the output value for a specific image quality measure.

Fig. 1 reveals that the quality of a locally smoothed (`LS`) image is better in comparison to the other two smoothing techniques (i.e. `GS` and `GS+LS`) for all IQM. GS deteriorates image quality quickly for increasing smoothing strength. Also for the combined method GS+LS the quality is found to be rather low due to the impact of GS. The quality of the LS images is better because we are smoothing only the patches around SIFT keypoints while other pixels are left untouched. As expected, when increasing the patch size in LS and GS+LS, the quality of the pro-

[1] Similar looking structures within an image may lead to an image incorrectly being classified as copy move forged image.

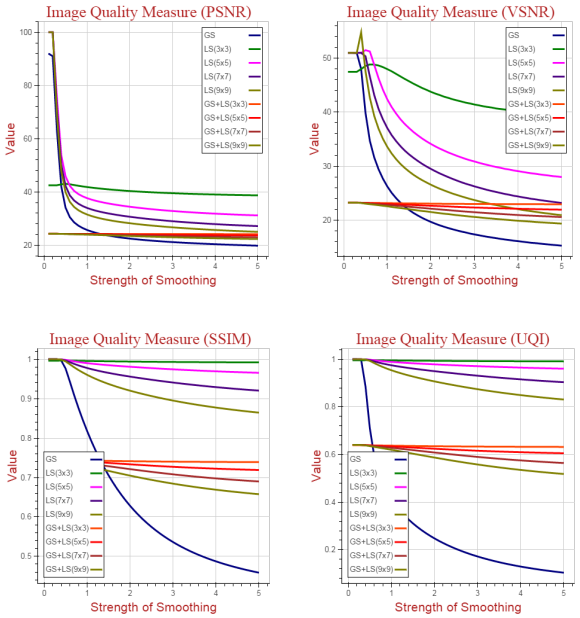


Figure 1: IQM Comparison among GS vs LS vs GS+LS

cessed images decreases.

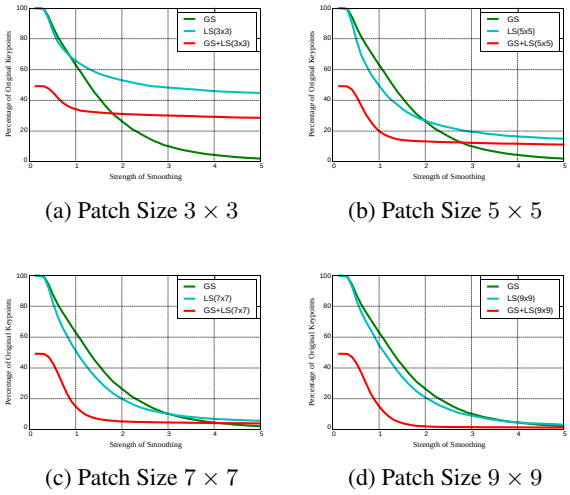
Table 1 displays IQM values for the CA.

Patch Sizes	PSNR	VSNR	UQI	SSIM
3x3 Patch	64.80	32.78	0.99	0.99
5x5 Patch	51.95	32.23	0.99	0.99
7x7 Patch	47.10	47.58	0.99	0.99
9x9 Patch	42.86	40.89	0.98	0.99

Table 1: IQM for CA.

For UQI as well as SSIM we notice almost no quality degradation by the CA, no matter which patch size is being used. For PSNR, CA is superior to all GS+LS variants and for almost all other settings except for extremely low smoothing strength. Finally, for VSNR, CA is again superior to all GS+LS variants and for all other techniques but LS with patchsize 3 for low smoothing strength. Overall, the quality obtained with the CA is very good, and only comparable to LS with patchsize 3, however, with all patch sizes considered.

But how effective are the smoothing-based methods in actually removing keypoints ? Contrasting to CA, in which all present keypoints are replaced by keypoint-free patches, smoothing does not guarantee that keypoints are actually removed. Fig. 2 illustrates the percentage of original keypoints which are still present after smoothing for increasing smoothing strength.



(a) Patch Size 3×3 (b) Patch Size 5×5

(c) Patch Size 7×7 (d) Patch Size 9×9

Figure 2: Share of retained keypoints: GS vs LS vs GS+LS

For larger patch sizes, GS and LS perform almost identically (which is clear considering the definition), while GS+LS is most effective in removing keypoints. For smaller patch sizes, GS is most effective for high smoothing strength, while GS+LS is best for low smoothing strength. LS is not very effective under these conditions.

When applying techniques for keypoint removal, new keypoints are being created, e.g. at the edge of the patches in CA, LS, and GS+LS. This is not desired, as these new keypoints might match to existing ones and thus aid the forensic analyst. Fig. 3 illustrates the creation of new, additional keypoints by showing the percentage of newly created ones. LS clearly introduces the lowest number of additional keypoints, and if the size of the smoothing patch is increased then also the number of new keypoints is also increased. The smoothing strength also plays a certain role: For weak smoothing, increasing the strength leads to more new keypoints, while after reaching a peak, a further increase of smoothing strength decreases the number of newly created keypoints. This effect is expected and most obvious for GS.

In Table 2, the percentage of newly created keypoints for CA is shown. Only LS with patchsize 3 gives better results, for all other techniques we notice higher percentages of newly created keypoints when comparing Fig. 3 to the values in Table 2.

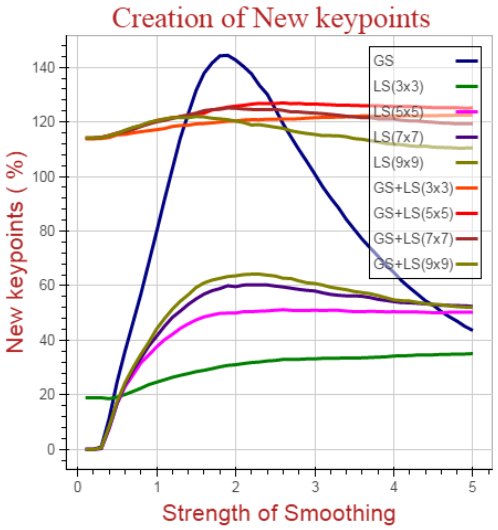


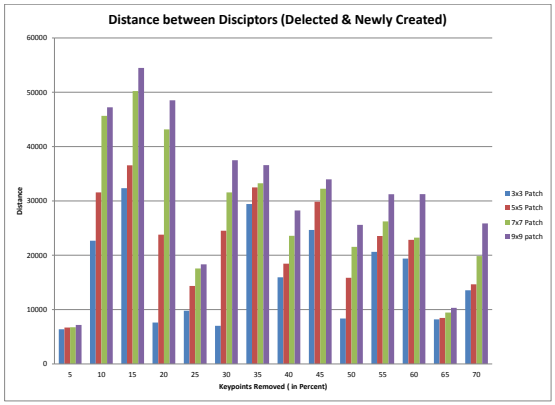
Figure 3: Creation of New Keypoints.

3x3 Patch	5x5 Patch	7x7 Patch	9x9 Patch
43.01%	39.76%	34.43%	32.21%

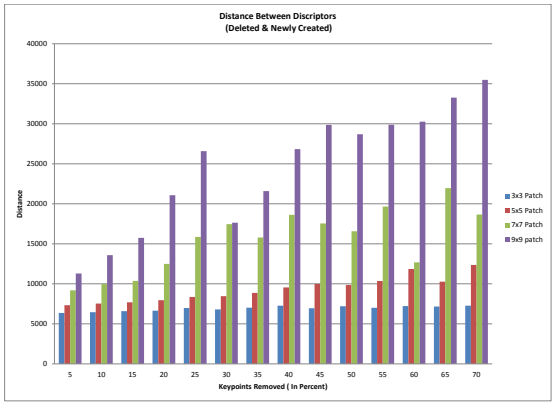
Table 2: Newly Generated Keypoints in CA.

When new keypoints are being generated, it is not their number that is most important. The aim of removing keypoints is compromised, if the newly generated ones are similar to the removed ones in terms of their SIFT descriptors. In this case, attacks might still be recovered by the forensic analyst even though keypoints have been removed. In Fig. 4 we plot the distance (squared Euclidean distance (SED)) of the SIFT descriptors describing removed and newly created ones. In particular, we compute SED between removed keypoints and their closest newly generated keypoints in terms of their descriptors. To avoid bias, we divide the result by the number of removed keypoints, as we display results in terms of increasing percentage of removed keypoints.

For the patch-based techniques, an increase of the patch size leads to higher SED, which is expected and desired. When increasing the percentage of removed keypoints, there is a tendency for increasing SED, except for LS and CA with smaller patch sizes. The largest SED values (which is the aim when removing keypoints) are seen for techniques involving GS (not shown) when a large share of all keypoints is being removed. CA clearly exhibits the lowest values, which means that the advantage of this approach in removing all keypoints is endangered by the creation of new keypoints which are close to the



(a) LS



(b) CA

Figure 4: Distance to newly created keypoints.

removed ones in terms of their SIFT descriptors.

After having analysed four different SIFT keypoint removal techniques with respect to different properties, we tested these methods in an actual copy move forgery scenario. The following definitions are employed:

- $TruePositive(TP)$: A true positive test result for a forged image is one that detects at least τ matching keypoint pairs.
- $FalseNegative(FN)$: A false negative test result for a forged image is one that detects at most $\tau - 1$ matching keypoint pairs.
- $TrueNegative(TN)$: A true negative test result for an image from the `BUILDING PHOTO` category is one that detects at most $\tau - 1$ matching keypoint pairs.

- $FalsePositive(FP)$: A false positive test result for an image from the BUILDING PHOTO category is one that detects at least τ matching keypoint pairs.

Based on these definitions, we are able to compute *precision*, *recall*, and *F1-score*. Recall, that the aim of the attacker is to disable the techniques of the forensic analyst. Thus, the attacker developing these techniques to counter SIFT keypoint based forensic techniques by removing keypoints aims for low TP (and low TN), as high FN makes the forensic analyst miss forged images and high FP confuses the analyst as many genuine images are determined as forgeries.

First, we computed SIFT keypoints and then for each keypoint we found the two nearest neighbours from all remaining keypoints using a K-d tree based on Euclidean distance d_1 and d_2 (where d_1 and d_2 are distances and d_1 corresponds to the closest neighbour), $T \in (0, 1)$. [13] and [11] suggested that there is a match only if $\frac{d_1}{d_2} < T$ holds. In these papers $T = 0.6$ but we looked into results for $T = 0.4$, $T = 0.5$, $T = 0.6$ and $T = 0.7$.



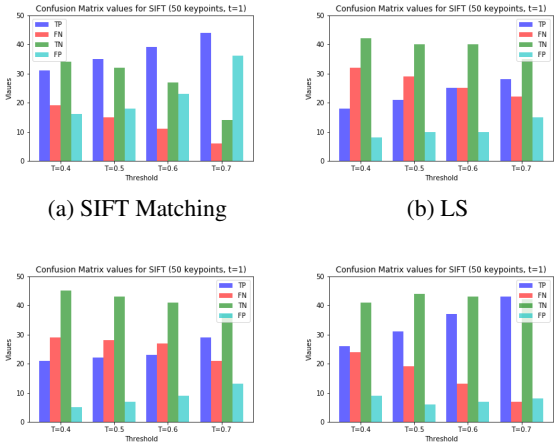
Figure 5: Forged Images



Figure 6: Original Images

Fig. 7 shows confusion matrices (i.e. the number of TP, FN, TN, FP) for using 50 keypoints, $\tau = 1$, for four different values of T, comparing copy move forgery detection without manipulating images, and with applying keypoint removal techniques LS, CA, and GS+LS. Patch size is set to 9x9 pixels in all patch-based techniques.

Overall, we observe that all three SIFT keypoint removal strategies work, i.e. they reduce significantly the number of TP. However, they increase also the number of TN, thus, the number of false positives is also reduced (which is not desired). When we



(a) SIFT Matching (b) LS

(c) CA (d) GS+LS

Figure 7: Copy Move Forgery Detection

compare the three removal strategies, GS+LS clearly has a higher number of TP, thus is least efficient and does not need to be considered further in this comparison. LS and CA are close, with slight advantages for LS, however, difficult to confirm in this visual representation.

When looking into recall and precision values for $\tau = 1, 2, 3$ and $T = 0.4, 0.5, 0.6, 0.7$ using 50, 100, and 200 keypoints (overall 36 configurations), we find precision(LS) $<$ precision(CA) in 33/36 cases, while recall(LS) $<$ recall(CA) in 20/36 cases. Therefore, overall, LS is clearly more effective in preventing to detect a copy move forgery as CA is. In terms of F1-score F1(LS) $\leq$ F1(CA) in 27/36 cases, which confirms the trend.

Table 3 shows precision, recall and F1-scores of the confusion matrices shown in Fig. 7. The cases in which LS delivers the best (lowest) results are underlined - we notice that this is also the clear majority within these result subsets.

		CA			LS			GS+LS		
τ	T	Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
1	0.4	0.81	0.42	0.55	0.69	0.36	0.47	0.86	0.60	0.71
1	0.5	0.76	0.44	0.56	0.68	0.42	0.52	0.81	0.62	0.70
1	0.6	0.72	0.46	0.56	0.71	0.50	0.57	0.84	0.74	0.79
1	0.7	0.69	0.58	0.63	0.65	0.56	0.60	0.84	0.86	0.85

Table 3: Comparison of keypoint removal techniques in terms of precision, recall, and F1-score.

4. Conclusion

Local smoothing (LS), as proposed in this paper, turns out to be more effective in preventing a detection of a copy move attack as compared to the col-

lage attack (CA). For the patch-size chosen in the comparison, the image quality is slightly superior for CA. GS and GS+LS as also proposed in literature are neither competitive in terms of maintained image quality nor in terms of preventing the copy move attack detection capability. When considering the ease of application, LS is clearly preferable, as CA requires the generation of a keypoint-free dictionary and a vector-quantisation like patch selection process, while LS only applies a local Gaussian smoothing. Overall, LS turns out to be a highly attractive alternative to SIFT keypoint removal techniques applied so far in literature.

References

[1] I. Amerini, M. Barni, R. Caldelli, and A. Costanzo. Counter-forensics of sift-based copy-move detection by means of keypoint classification. *EURASIP Journal on Image and Video Processing*, 2013(1):18, 2013.

[2] I. Amerini, M. Barni, R. Caldelli, and A. Costanzo. Removal and injection of keypoints for sift-based copy-move counter-forensics. *EURASIP Journal on Information Security*, 2013(1):8, 2013.

[3] I. Amerini, F. Battisti, R. Caldelli, M. Carli, and A. Costanzo. Exploiting perceptual quality issues in countering sift-based forensic methods. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2664–2668. IEEE, 2014.

[4] E. Ardizzone, A. Bruno, and G. Mazzola. Detecting multiple copies in tampered images. In *2010 IEEE International Conference on Image Processing*, pages 2117–2120. IEEE, 2010.

[5] E. Ardizzone, A. Bruno, and G. Mazzola. Copy–move forgery detection by matching triangles of keypoints. *IEEE Transactions on Information Forensics and Security*, 10(10):2084–2094, 2015.

[6] V. Christlein, C. Riess, J. Jordan, C. Riess, and E. Angelopoulou. An evaluation of popular copy-move forgery detection approaches. *IEEE Transactions on Information Forensics and Security*, 7(6):1841–1854, 2012.

[7] A. Costanzo, I. Amerini, R. Caldelli, and M. Barni. Forensic analysis of sift keypoint removal and injection. *IEEE Transactions on Information Forensics and Security*, 9(9):1450–1464, 2014.

[8] D.-T. Dang-Nguyen, C. Pasquini, V. Conotter, and G. Boato. Raise: a raw images dataset for digital image forensics. In *Proceedings of the 6th ACM Multimedia Systems Conference*, pages 219–224. ACM, 2015.

[9] T.-T. Do, E. Kijak, T. Furon, and L. Amsaleg. Deluding image recognition in sift-based cbir systems. In *Proceedings of the 2nd ACM workshop on Multimedia in forensics, security and intelligence*, pages 7–12. ACM, 2010.

[10] C.-Y. Hsu, C.-S. Lu, and S.-C. Pei. Secure and robust sift. In *Proceedings of the 17th ACM International Conference on Multimedia*, pages 637–640. ACM, 2009.

[11] H. Huang, W. Guo, and Y. Zhang. Detection of copy-move forgery in digital images using sift algorithm. In *Pacific-Asia Workshop on Computational Intelligence and Industrial Application, 2008. PACIIA'08.*, volume 2, pages 272–276. IEEE, 2008.

[12] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, November 2004.

[13] B. Mahdian and S. Saic. Detection of copy–move forgery using a method based on blur moment invariants. *Forensic Science International*, 171(2-3):180–189, 2007.

[14] MATLAB. *version 8.3.0.532 (R2014a)*. The MathWorks Inc., Natick, Massachusetts, 2014.

[15] M. MuX. *MeTriX MuX version 1.1*. 2014.

[16] J. Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in Large Margin Classifiers*, 10(3):61–74, 1999.

[17] M. Saleem. A key-point based robust algorithm for detecting cloning forgery. In *IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, volume 4, pages 2775–2779, 2014.

[18] G. Schäfer and M. Stich. UCID - an uncompressed colour image database. *Proc. SPICE. Storage and Retrieval Methods and Applications for Multimedia*, 11(1):472–480, 2004.

[19] H. Sencar and N. M. (Eds.). *Digital Image Forensics: There is more to a picture than meets the eye*. Springer Verlag, 2012.

[20] A. Vedaldi and B. Fulkerson. Vlfeat: An open and portable library of computer vision algorithms. In *Proceedings of the 18th ACM International Conference on Multimedia*, MM '10, pages 1469–1472, New York, NY, USA, 2010. ACM.

[21] M. Zandi, A. Mahmoudi-Aznaveh, and A. Mansouri. Adaptive matching for copy-move forgery detection. In *2014 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 119–124. IEEE, 2014.

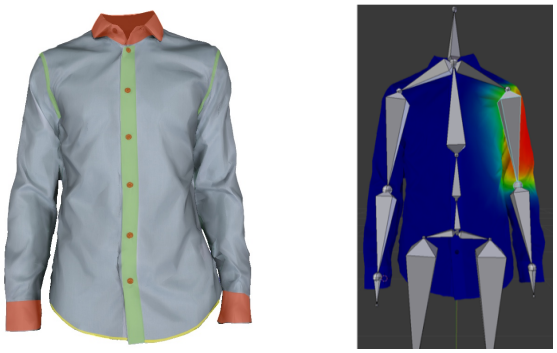
[22] D. Zhang and G. Lu. Evaluation of similarity measurement for image retrieval. In *International Conference on Neural Networks and Signal Processing, 2003. Proceedings of the 2003*, volume 2, pages 928–931. IEEE, 2003.

AUTOMATED GENERATION OF 3D GARMENTS IN DIFFERENT SIZES FROM A SINGLE SCAN

Stefan Hauswiesner, Philipp Grasmug
Reactive Reality
{hauswiesner,grasmug}@reactivereality.com



Figure 1: Results of the method.



(a) (b)

Figure 2: Sizing the mesh. a) semantic classification map. Red parts do not scale with size, green parts scale along a single dimension. b) sizing an upper arm. The color coding shows the body part association.

Abstract. *We describe a method to generate additional sizes of a garment from a single scanned size and grading tables. The method helps retailers and manufacturers to efficiently capture their entire product range, which in turn enables advanced AR applications such as virtual fashion try-on.*

1. Introduction

Online fashion retailers need 3D models of their entire product range to enable advanced e-commerce applications, such as 3D viewing and virtual try-on. These retailers usually have a high number of items and the product range changes frequently. Therefore, to obtain 3D models for their entire product catalog, manual modeling is not a feasible approach.

3D reconstruction through photogrammetry is more efficient and photo-realistic. However, retailers want to avoid the overhead of scanning multiple sizes of a single garment. This document describes a different approach which algorithmically generates the different sizes from a single 3D reconstructed model and the garment's grading tables, and thereby increases the scalability of photogrammetry. Figure 1 shows results.

2. Related Work

Previous work in garment modeling generates sizes by adapting a garment to a target body, as opposed to sizing tables [1]. Other machine-learning-based approaches enable decomposition and assembly of new garments but do not allow resizing [2] or rely on templates [4], which inherently limits these approaches to known shapes.

3. Method

The challenge of size synthesis is that garments do not scale uniformly. For example, going from size "Medium" to size "Large", the scale factor for the length of the sleeves is different from the factor for the circumference of the sleeve. The way a garment's parts scale is described by a grading table. We use this information to adjust the geometry of the model for the distinct sizes.

Furthermore, the fabric of the garment is not

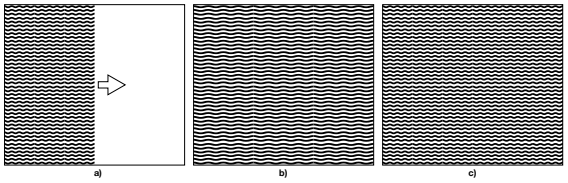


Figure 3: Increasing the size of a texture patch (a) by scaling (b) or repetition (c).

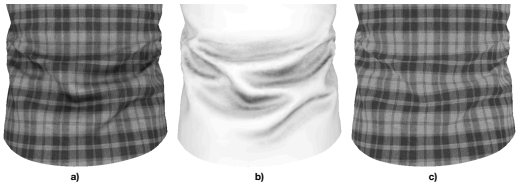


Figure 4: Texture decomposition of (a) into illumination (b) and material (c).

scaled uniformly but repeated. Knowledge of the used materials is needed to simulate this behavior. Elements like buttons or pockets also do not scale, or only under certain constraints (e.g., seams or zippers scale in one direction). Prints on a garment usually also scale independently from the pattern of the fabric. The behavior usually cannot be described by a set of global rules. Therefore, the proposed system provides a way to adjust the scaling behavior for each element independently.

3.1. Input

The method takes a 3D garment model created through photogrammetry and a size chart as its input. The garment model consists of a mesh and a mapped texture. A parametric body model consisting of a pose and a shape description is registered to the 3D garment model. The measurements of the grading table are associated with the parametric body model in the form of edge paths.

3.2. Semantic Region Segmentation

First, the garment mesh and its texture are input to a machine learning algorithm which assigns a semantic meaning to each texel (e.g. collar, seam, button, etc.). Moreover, the same algorithm labels background and mannequin texels for removal. The map's semantic meaning can be transferred to the mesh's faces and vertices through texture mapping.

3.3. Sizing the Mesh

The grading table describes how different elements like sleeves, collars, legs, etc. scale between the sizes. Each measurement is associated with an edge path in the parametric body model. These paths are projected onto the garment's mesh. The actual scaling transformation is performed through Laplacian Mesh Processing [3]. Parts which should not scale obtain high regularization weights. The edge path lengths act as the data terms of the desired transformation. See Figure 2.

3.4. Sizing the Texture

Simply scaling a garment's mesh and texture based on the grading table and the parametric body model is not enough because fabrics are not stretched but rather more of the fabric is used (Figure 3). This is achieved by repeating the texture instead of scaling. The pattern repetition is aligned with the sewing/cutting lines of the garment, which are derived from the parametric body like measurement paths. Finally, the texture needs to be preprocessed to separate the material's diffuse color from large scale lighting effects, such as wrinkles which should not be repeated. Figure 4 shows the decomposition.

4. Conclusion

We have shown a method to generate additional sizes of a garment from a single scanned size and grading tables. The method helps retailers and manufacturers to efficiently capture their entire product range, e.g. for virtual fashion try-on. Moreover, this work demonstrates how to overcome a major limitation of photogrammetry: the ability to create 3D models of items which are not available for scanning.

References

[1] R. Brouet, A. Sheffer, L. Boissieux, and M.-P. Cani. Design preserving garment transfer. *ACM Trans. Graph.*, 31(4), July 2012.

[2] L. Liu, Z. Su, X. Fu, L. Liu, R. Wang, and X. Luo. A data-driven editing framework for automatic 3d garment modeling. *Multimedia Tools Appl.*, 76(10):12597–12626, May 2017.

[3] O. Sorkine. Laplacian Mesh Processing. In Y. Chrysanthou and M. Magnor, editors, *Eurographics 2005 - State of the Art Reports*. The Eurographics Association, 2005.

[4] Y. Xu, S. Yang, W. Sun, L. Tan, K. Li, and H. Zhou. Virtual garment using joint landmark prediction and part segmentation. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 1247–1248, 2019.