Linköping Studies in Science and Technology
Dissertations, No. 2151

Learning Representations for Segmentation and Registration

Felix Järemo Lawin

Linköping Studies in Science and Technology
Dissertations, No. 2151

Learning Representations for Segmentation and Registration

Felix Järemo Lawin

Linköping University
Department of Electrical Engineering
Computer Vision Laboratory
SE-581 83 Linköping, Sweden

Linköping 2021

This work is licensed under a Creative Commons Attribution 4.0 International License.

<https://creativecommons.org/licenses/by/4.0/>

Edition 1:1

© Felix Järemo Lawin, 2021

ISBN 978-91-7929-623-0

ISSN 0345-7524

URL <http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-176054>

Published articles have been reprinted with permission from the respective copyright holder.

Typeset using X_YTeX

Printed by LiU-Tryck, Linköping 2021

POPULÄRVETENSKAPLIG SAMMANFATTNING

Inom datorseende är målsättningen att automatiskt kunna extrahera och modellera högnivårepresentationer från visuella sensormätningar som bilder, videosekvenser och 3D-punkter. Visuella data är ofta högdimensionella, brusiga och oregelbundna, vilket leder till många utmaningar vid modellering. I den här avhandlingen presenteras forskningsarbeten som adresserar sådana utmaningar inom ett flertal olika datorseendeproblem.

Först undersöks fasuppackningsproblemet för multifrekvensbaserad amplitudmodulerad time-of-flight (ToF). ToF är en teknik som används i djupkameror, vilka har många applikationer inom 3D rekonstruktion och gestigenkänning. Amplitudmodulerad time-of-flight kan generera noggranna mätningar, men samtidigt introducera tvetydigheter i djupskattningarna. I avhandlingen presenteras en metod för att lösa problemet med tvetydigheterna genom att skatta sannolikheterna för multipla hypoteser för djupet. Detta uppnås genom att utföra en kärntäthetsskattning över alla hypoteser i en omgivning runt alla pixlar i djupbilden. Hypotesen med den högsta skattade sannolikheten väljs därefter ut som djupskattning. Den här metoden leder till en förbättrad kvalitet på djupbilderna, samt ökar den effektiva räckvidden för djupkameror i både inomhus och utomhusmiljöer.

Därefter adresseras punktmolnsregistrering, som är problemet att hitta den korrekta transformationen mellan olika överlappande punktmoln. Robust registrering är fundamentalt för flertal olika problem, som till exempel 3D-rekonstruktion och estimering av objektposition. Ett första bidrag i avhandlingen är en metod för att hantera täthetsvariationer i punktmolnen. Metoden använder sig av en probabilistisk modell av en latent sannolikhetsfördelning som representerar den underliggande strukturen i den observerade scenen. Både modellen och registreringsparametrarna optimeras med väntevärdes- maximerings-algoritmen. Vidare introducerar avhandlingen en metod för att integrera djupa särdrag i registreringsramverket. Det visar sig att de djupa särdragen förbättrar registreringsmetoden, både i termer av noggrannhet och robusthet. Ytterligare förbättringar uppnås genom att lära sig särdragsrepresentationer genom att minimera registreringsfel.

I avhandlingen presenteras även en metod för 3D-punktmolnssegmentering. Eftersom scenmodeller ofta representeras av 3D-punktmätningar, är segmentering av dessa viktigt för generell scenförståelse. Inlärning av modeller för segmentering kräver stora mängder etiketterad data, vilket är dyrt och tidskrävande att skapa. Metoden som presenteras i avhandlingen undviker detta problem genom att projicera punkterna i virtuella kameror och rendera 2D bilder. Metoden kan därmed utnyttja högpresterande tekniker baserade på djupa faltningsnätverk för bildsegmentering och minskar därför även behovet av 3D data med sanningsetiketter.

Slutligen adresseras video objektsegmentering (VOS), där målet är att följa och segmentera specifika objekt i en videosekvens. För att uppnå hög prestanda, behövs en robust objektmodell som kan anpassas till olika scenarier och förändringar i utseende. Den stora utmaningen består i att detta måste uppnås med hjälp av enbart en bild med given segmentering i videosekvensen. För att hantera utmaningarna inom VOS, presenterar den här avhandlingen en parametrisk målobjektetsmodell, optimerad för att prediktera en maskfunktion av målobjektet. Modellen integreras i ett djupt neuralt nätverk, där den vägleder ett segmenterings nätverk att prediktera målobjektssegment. Det djupa nätverket tränas för att maximera segmenteringsnoggrannheten på ett videosegmenteringsdataset med givna segmenteringar. Det visas även att samtidig träning av hela nätverket medför att metoden kan lära sig att ta fram en maskfunktion till modellen som maximerar den slutgiltiga segmenteringsnoggrannheten.

ABSTRACT

In computer vision, the aim is to model and extract high-level information from visual sensor measurements such as images, videos and 3D points. Since visual data is often high-dimensional, noisy and irregular, achieving robust data modeling is challenging. This thesis presents works that address challenges within a number of different computer vision problems.

First, the thesis addresses the problem of phase unwrapping for multi-frequency amplitude modulated time-of-flight (ToF) ranging. ToF is used in depth cameras, which have many applications in 3D reconstruction and gesture recognition. While amplitude modulation in time-of-flight ranging can provide accurate measurements for the depth, it also causes depth ambiguities. This thesis presents a method to resolve the ambiguities by estimating the likelihoods of different hypotheses for the depth values. This is achieved by performing kernel density estimation over the hypotheses in a spatial neighborhood of each pixel in the depth image. The depth hypothesis with the highest estimated likelihood can then be selected as the output depth. This approach yields improvements in the quality of the depth images and extends the effective range in both indoor and outdoor environments.

Next, point set registration is investigated, which is the problem of aligning point sets from overlapping depth images or 3D models. Robust registration is fundamental to many vision tasks, such as multi-view 3D reconstruction and object pose estimation for robotics. The thesis presents a method for handling density variations in the measured point sets. This is achieved by modeling a latent distribution representing the underlying structure of the scene. Both the model of the scene and the registration parameters are inferred in an Expectation-Maximization based framework. Secondly, the thesis introduces a method for integrating features from deep neural networks into the registration model. It is shown that the deep features improve registration performance in terms of accuracy and robustness. Additionally, improved feature representations are generated by training the deep neural network end-to-end by minimizing registration errors produced by our registration model.

Further, an approach for 3D point set segmentation is presented. As scene models are often represented using 3D point measurements, segmentation of these is important for general scene understanding. Learning models for segmentation requires a significant amount of annotated data, which is expensive and time-consuming to acquire. The approach presented in the thesis circumvents this by projecting the points into virtual camera views and render 2D images. The method can then exploit accurate convolutional neural networks for image segmentation and map the segmentation predictions back to the 3D points. This also allows for transferring learning using available annotated image data, thereby reducing the need for 3D annotations.

Finally, the thesis explores the problem of video object segmentation (VOS), where the task is to track and segment target objects in each frame of a video sequence. Accurate VOS requires a robust model of the target that can adapt to different scenarios and objects. This needs to be achieved using only a single labeled reference frame as training data for each video sequence. To address the challenges in VOS, the thesis introduces a parametric target model, optimized to predict a target label derived from the mask annotation. The target model is integrated into a deep neural network, where its predictions guide a decoder module to produce target segmentation masks. The deep network is trained on labeled video data to output accurate segmentation masks for each frame. Further, it is shown that by training the entire network model in an end-to-end manner, it can learn a representation of the target that provides increased segmentation accuracy.

Acknowledgments

I could not have made it through my Ph.D. studies without the help of my colleagues at CVL. First and foremost I would like to thank my supervisor Per-Erik Forssén for his friendliness and all the valuable guidance. Throughout, he has always been encouraging and open to discuss ideas. I have also enjoyed our supervision meetings, and often certain information has come to light during our discussions. Next, I want to thank my co-supervisor Michael Felsberg who first took me in as a Ph.D. student, or should I say, tricked me into it. Either way, I am glad that he gave me the chance at perusing a Ph.D. degree. I am very thankful for his support and for learning from his knowledge and experience. To both my supervisors I want to express my gratitude for giving me the freedom and the encouragement to explore different topics and ideas. I also want to thank all my collaborators and co-authors for sharing their brilliant ideas and for our interesting discussions. Martin Danelljan for his notation, for inviting me into exciting research projects and for pushing me into taking time-consuming math courses. Fahad Shahbaz Khan for his encouraging attitude and spreading of positive energy during the darkest deadline hours. Andreas Robinson for sharing his programming skills and for patiently helping me with software installations and computer issues. Goutam Bhat for his awesome figures, ability to improve the results and our discussions on football. I am very proud of what we achieved in our joint works. Moreover, I want to thank all of CVL, including past and current members, for the friendly atmosphere at the lab and all the interesting discussions during coffee breaks. I further want to thank Zahra for arranging barbecues and Mikael for hosting board gaming sessions. Hannes, Abdo and Emanuel for their nice company while sharing offices with me. I am also grateful to everyone who joined me for late lunch, even if the only acceptable option was the bad house. Special thanks go to Per-Erik, Gustav, Abdo, and Andreas for helping out with the proofreading of the thesis manuscript. Lastly, I want to thank my family and all my friends for all the support and good times.

Contents

Abstract	iii
Acknowledgments	v
Contents	vii
I Background	1
1 Introduction	3
1.1 Representations and learning	3
1.2 3D perception and registration	5
1.3 Segmentation	8
1.4 Outline	9
1.5 Included Publications	10
1.6 Additional Publications	15
2 3D perception and registration	17
2.1 Depth cameras	17
2.2 Time-of-flight ranging	19
2.3 Point set registration	22
3 Segmentation	37
3.1 Image Segmentation	37
3.2 3D Point Set Semantic Segmentation	40
3.3 Video object segmentation	44
4 Concluding remarks	57
Bibliography	61

II	Publications	75
Paper A	Efficient Multi-Frequency Phase Unwrapping using Kernel Density Estimation	77
Paper B	Density Adaptive Point Set Registration	103
Paper C	Registration Loss Learning for Deep Probabilistic Point Set Registration	123
Paper D	Deep Projective 3D Semantic Segmentation	139
Paper E	Learning Fast and Robust Target Models for Video Object Segmentation	155
Paper F	Learning What to Learn for Video Object Segmentation	171

Part I

Background

Introduction

In computer vision, the objective is to automate the extraction and modeling of high-level information from visual sensor data such as images, video streams and 3D measurements. The visual information can be used to model the observed environment. This involves generating 3D reconstructions, segmentation of the data into different objects and structures, and recognizing events and actions. Such models and representations are crucial in applications for autonomous systems, hence computer vision plays a central role in the industries of robotics, autonomous driving and medical treatment technologies. Nonetheless, there are many challenges to be addressed within the field to improve the performance of methods and their ability to generalize to different scenarios. This thesis presents methods that address computer vision problems within time-of-flight ranging for depth cameras, point set registration, 3D segmentation and video object segmentation.

1.1 Representations and learning

In a general sense, methods for computer vision aim at finding and modeling patterns in visual data. Strong pattern recognition further allows us to decompose the data into meaningful representations, like classes, geometric structures or events. For low-dimensional data, e.g. one-dimensional depth measurements, it is often sufficient to employ simple mixture models or even parameter-free density estimators [83, 94]. These can efficiently estimate the underlying probability density function of the observed data samples and further be employed in e.g. denoising, classification and registration tasks. Higher-dimensional data on the other hand, such as images, require more descriptive models.

A powerful method for modeling images is learning deep convolutional neural networks [64, 35, 65], which we for brevity hereon will denote as convnets. Roughly described, convnets are parametric models consisting

Figure 1.1: Illustration of a feature extractor convnet and feature maps extracted from an image from the DAVIS dataset [88]. In this case, the feature extractor is a ResNet50 model [45] that has been trained for image classification. A selection of channels is provided from the outputs of the shallow layer 1 and the deep layer 3. In the layer 1 features, we observe high activations around blobs and edges in the background and foreground. In contrast, layer 3 outputs high activations on specific objects and regions of the image.

of function compositions that are often denoted as layers. Typically, these layers comprise simple operators like convolutional filters, standard linear operators, point-wise non-linear activation functions, or self-attention modules [110], but they compose a highly descriptive and powerful function. We further note that the layers can also include more complex processes and even optimization algorithms [41, 5, 6].

A conventional convnet takes images as input, passes them through its layers and outputs a prediction. Each layer transforms the outputs from previous layers, producing image features with abstraction levels that increase towards the output representation. As an example, in convnets trained for image classification, we observe that the initial layers produce low-level representations that highlight traits such as edges, colors and blobs. In contrast, the deeper layers typically produce features that encode specific objects and regions corresponding to semantic classes (see Figure 1.1 for an illustrative example). The features from the different layers can further be decoded into class predictions and image segmentations, or be used in algorithms for regression tasks, geometric modeling and tracking. Although the success of convnets has been prominent for many computer vision tasks in recent years, there still exist many challenges in the learning of discriminative features and robust classifiers. Important to note is that the performance of methods highly depends on the properties of the image feature representations.

The parameters of convnets are learned through optimizing an objective function, generally by using back-propagation [95] and stochastic gradient descent [92]. Whenever ground truth labels for the data are available, the

learning can be supervised with an objective that equals or resembles the accuracy measure. Unfortunately, annotating ground truth labels often requires significant amounts of manual labor, making it infeasible to construct sufficiently large datasets for learning the parameters without invoking severe overfitting issues. For this reason, it is common to perform transfer learning through pre-training of feature extractor convnets using weaker annotations. This allows for learning convnets that produce initial feature representations that can be reused for a variety of different tasks. Many methods rely on pre-training since they are unable to train the feature extractors with the other parts of the network model jointly. This can be caused by the network model not being fully differentiable or practical reasons like memory consumption constraints during network training.

The benefit from transfer learning is demonstrated in many scenarios, including segmentation tasks [71] and descriptor learning for registration [12]. It can both reduce training time and the required amount of annotated data. However, the pre-trained feature extractors are optimized for an auxiliary task, which may produce sub-optimal representations for the main objective. It is generally preferable to train the entire network end-to-end, i.e. learning the parameters of all network components jointly by minimizing an objective over the outputs of the network. Such methods [6, 5, 62] can in theory learn feature representations that maximize the performance for the main task.

In the works presented in this thesis, we adopt and develop deep network architectures and apply them for segmentation tasks and point set registration. In these approaches, we employ pre-trained networks to produce powerful representations for images and 3D data. Specifically, we develop optimization-based frameworks and investigate how these can leverage discriminative features to make robust predictions. We further demonstrate improved performance when the optimization-based frameworks are integrated into end-to-end trainable models. The following sections describe the problems investigated in the thesis and provide an introduction to the proposed methods.

1.2 3D perception and registration

Autonomous systems that interact in the real world face challenges in navigation, obstacle avoidance, and object manipulation. Therefore, it is crucial to maintain a precise representation of the observed 3D scene, further requiring accurate 3D perception and robust modeling techniques.

In this thesis, we tackle problems within depth ranging and 3D point set registration for 3D reconstruction. In 3D reconstruction, the objective is to model the 3D structure of the observed scene given a sequence of images acquired from different camera positions. Such a model can include a 3D map of the environment along with the locations and orientations of the cameras.

It can be generated using structure from motion with sequences of RGB images or by aligning sequences of depth images. This thesis investigates depth imaging and methods for aligning these to reconstruct the 3D scene. Compared to structure from motion with RGB images, a 3D reconstruction based on depth images generally gives denser and more accurate 3D models.

Time-of-flight ranging

Depth imaging has many applications in 3D modeling, mapping and gesture recognition. Following the recent development of cheap depth cameras, depth ranging has become more available to the research community. This has contributed to improvements within methods for 3D reconstruction and robotics tasks.

A popular approach for measuring depth in depth cameras is to apply time-of-flight ranging with amplitude modulation [33, 43]. In this method, the phase shift of an emitted sinusoidal light signal is measured and subsequently converted into a depth measurement. However, the periodicity of the signal leads to a depth ambiguity. To address this problem, Paper A of this thesis proposes a method for phase unwrapping using multiple modulation frequencies. In the proposed method, we estimate the distribution of possible depths over multiple depth hypotheses. Further robustness to sensor noise is achieved by computing a weighted kernel density estimate in a spatial neighborhood of each pixel in the image. The final depth can then be chosen as the one with the highest estimated density. We demonstrate that this approach can improve the accuracy of depth cameras and extending their effective range in both indoor and outdoor environments.

Point Set Registration

3D data, such as object shapes and scene measurements, are often represented as point sets, i.e. arrays of 3D coordinates. These can be generated from computer tomography scans or by out projecting depth images from RGB-D cameras and Lidar scanners using their projective geometry. Point set registration is the task of aligning point sets from overlapping images by finding the relative transformations between them. It can further be used to fuse sequences of depth images into a 3D map of the scene and for finding the pose of objects.

In Paper B and Paper C, we investigate probabilistic approaches for registration based on Expectation-Maximization (EM) [47, 82, 49, 28, 18]. These approaches represent the scene as a mixture model given the observed 3D points. The parameters of the model and the relative transformations are optimized jointly using a maximum likelihood formulation and EM.

While previous EM based approaches have shown promising performance, they struggle in the presence of variations in the density of 3D points.

Figure 1.2: Visualization of four 3D point sets from the Kitti odometry dataset [36]. Left shows the point sets before registration with the initial transformation parameters. Right shows the resulting registered point sets after we have applied the algorithm from Paper C.

Such variations are caused by the sampling patterns of depth cameras, the distance to the camera location, and the local geometry of the observed surfaces. EM based methods struggle in these scenarios since they implicitly assume that the local densities of the point sets are constant throughout the observed scene. In Paper B, we address this problem by estimating the underlying distribution of the scene rather than directly model the distribution of the points. Specifically, we find the relative transformation of the point sets by reformulating EM to minimize the Kullback-Leibner divergence between the model and a latent scene distribution. As a result, we can perform robust registration despite the presence of severe density variations.

To achieve further robustness, Paper C proposes an approach to incorporate deep point set features into the probabilistic model of the scene. We demonstrate that the features significantly improve the robustness and accuracy of the registration method. Further, due to the probabilistic formulation, the full registration process is differentiable. This allows for training deep networks in an end-to-end manner to generate the feature representations of the point sets. To this end, we design a loss function based on registration errors, computed by deploying our method on datasets of point set sequences and comparing the predicted transformations with the ground truth. We learn the parameters of the network by back-propagating the gradients of the loss through the EM iterations. Figure 1.2 shows an example of aligned point sets using our approach in Paper C.

1.3 Segmentation

Segmentation is one of the main topics in computer vision. In segmentation problems, the task is to classify each point or region in a data sample. For images and videos, that corresponds to classifying each pixel, but the task also generalizes to the point-wise classification of unstructured point sets. Segmentation provides a compact, yet descriptive and high-level representation of the data. It is highly useful for scene understanding, and fundamental to many applications within robotics and autonomous driving. In image segmentation, state-of-the-art methods generally employ convnets, trained to predict segmentation score maps over the pixel locations of the input image. In this thesis, we adopt such networks for 3D point set segmentation and video object segmentation.

3D Point Set Segmentation

In section 1.2 above, we first discussed depth image ranging, followed by the registration of 3D point sets. As a next step, we are interested in further analysis of 3D data and reasoning over the content of the observed scene. One of the major topics within 3D scene understanding is semantic segmentation, which is the task of automatically assigning each 3D point with a semantic label from a set of pre-defined classes.

Compared to regular image segmentation tasks, 3D point set segmentation has some additional challenges. Firstly, point sets are sparse and unstructured, obstructing the direct use of standard convnet architectures applied on regular grids. Secondly, manual semantic labeling of 3D point sets is very time-consuming, even compared to labeling of image segmentation data. This reduces the amount of available training data, complicating the training of deep networks.

In Paper D, we develop a framework for 3D semantic segmentation that exploits the advantages of deep image segmentation approaches. In our method, the point set is first projected and rendered into a set of synthetic images using a splatting technique. These images can be segmented using regular image segmentation networks. The resulting pixel-wise segmentation scores are then re-projected into the point sets. Finally, the semantic label for each point is then obtained by fusing scores over the different views. Importantly, the segmentation network can leverage pre-training on image segmentation datasets, thus requiring less 3D data with labels.

Video Object Segmentation

Video object segmentation (VOS) is the problem of performing pixel-wise classification of specific target objects throughout a video sequence. It can be regarded as a generalization of point tracking and visual object tracking,

with the extension that the target can take any shape. VOS is very useful for video and scene understanding, and has many applications within surveillance, autonomous driving, video editing and robotic systems.

This thesis focuses on semi-supervised VOS, where we assume to be given the masks of the targets in the initial video frame. The task is then to track and segment the target objects in all subsequent frames. Despite the limited information in the initial frame, accurate VOS methods have to adapt to the targets while handling appearance changes, occlusions, and similar objects in the background.

In Paper E and Paper F, we create convnets for VOS that integrates target-specific appearance models for adapting the network to the targets in current video sequence. While prior works employ target feature embeddings, attention and generative modeling, our target model composes a compact parametric operator. The operator parameters are optimized to minimize an objective function based on the errors between target predictions and the target mask. In the subsequent frames, we use the learned target model to predict a coarse representation of the target. This representation is then decoded into a high-quality segmentation mask by a segmentation network. Unlike many previous methods, our target model can operate without feature embeddings specifically learned for the task. Instead, it can be employed directly on features extracted by networks that have been pre-trained for image classification. As a result, our method can achieve impressive VOS performance despite limited amounts of VOS training data.

In Paper F, we further improve the method by employing an end-to-end learning strategy of our VOS network. This includes learning the parameters of a feature extractor. Additionally, our end-to-end learning formulation allows us to generalize the target representation in the network. Instead of using the single-channel target mask as the label in the objective of the target model optimization, we learn a richer multi-channel label encoding. We demonstrate that this learned target representation provides useful information for guiding the VOS network into predicting accurate target masks. Lastly, we extend our method to handle bounding box annotations of the target in the first frame, enabling less demanding supervision during inference.

1.4 Outline

The thesis is divided into two parts. Part I, provides an overview of the investigated problems and the contributions of the thesis. It contains the two chapters of *3D perception and registration* and *segmentation*. The first chapter discusses the topics of time-of-flight and point set registration. The second chapter of Part I discusses 3D point set segmentation and video object segmentation. Part II contains the full versions of the publications presented in the thesis, including supplementary materials.

1.5 Included Publications

The publications included in the thesis are listed below, along with abstracts and author contributions.

Paper A: Efficient Multi-Frequency Phase Unwrapping Using Kernel Density Estimation

Felix Järemo Lawin, Per-Erik Forssén, and Hannes Ovrén. “Efficient multi-frequency phase unwrapping using kernel density estimation”. In: *European Conference on Computer Vision (ECCV), Amsterdam, 2016*, pp. 170-185, Springer, Cham. doi: 10.1007/978-3-319-46493-0_11.

Abstract: In this paper we introduce an efficient method to unwrap multi-frequency phase estimates for time-of-flight ranging. The algorithm generates multiple depth hypotheses and uses a spatial kernel density estimate (KDE) to rank them. The confidence produced by the KDE is also an effective means to detect outliers. We also introduce a new closed-form expression for phase noise prediction, that better fits real data. The method is applied to depth decoding for the Kinect v2 sensor, and compared to the *Microsoft Kinect SDK* and to the open source driver *libfreenect2*. The intended Kinect v2 use case is scenes with less than 8m range, and for such cases we observe consistent improvements, while maintaining real-time performance. When extending the depth range to the maximal value of 18.75m, we get about 52% more valid measurements than *libfreenect2*. The effect is that the sensor can now be used in large depth scenes, where it was previously not a good choice.

Author’s contribution: The author developed the methods in collaboration with Per-Erik Forssén, conducted the experiments and was the main contributor to the manuscript.

Paper B: Density Adaptive Point Set Registration

Felix Järema Lawin, Martin Danelljan, Fahad Shahbaz Khan, Per-Erik Forssén, and Michael Felsberg. “Density adaptive point set registration”. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3829–3837. doi: 10.1109/CVPR.2018.00403.

Abstract: Probabilistic methods for point set registration have demonstrated competitive results in recent years. These techniques estimate a probability distribution model of the point clouds. While such a representation has shown promise, it is highly sensitive to variations in the density of 3D points. This fundamental problem is primarily caused by changes in the sensor location across point sets. We revisit the foundations of the probabilistic registration paradigm. Contrary to previous works, we model the underlying structure of the scene as a latent probability distribution, and thereby induce invariance to point set density changes. Both the probabilistic model of the scene and the registration parameters are inferred by minimizing the Kullback-Leibner divergence in an Expectation Maximization based framework. Our density-adaptive registration successfully handles severe density variations commonly encountered in terrestrial Lidar applications. We perform extensive experiments on several challenging real-world Lidar datasets. The results demonstrate that our approach outperforms state-of-the-art probabilistic methods for multi-view registration, without the need of re-sampling.

Author’s contribution: The author conducted the experiments, was the main contributor to the manuscript and developed the methods in collaboration with Martin Danelljan.

Paper C: Registration Loss Learning for Deep Probabilistic Point Set Registration

Felix Järema Lawin and Per-Erik Forssén. “Registration Loss Learning for Deep Probabilistic Point Set Registration”. In: *2020 International Conference on 3D Vision (3DV)*, 2020, pp. 563-572. doi: 10.1109/3DV50981.2020.00066.

Abstract: Probabilistic methods for point set registration have interesting theoretical properties, such as linear complexity in the number of used points, and they easily generalize to joint registration of multiple point sets. In this work, we improve their registration performance to match state of the art. This is done by incorporating learned features, by adding a von Mises-Fisher feature model in each mixture component, and by using learned attention weights. We learn these jointly using a registration loss learning strategy (RLL) that directly uses the registration error as a loss, by back-propagating through the registration iterations. This is possible as the probabilistic registration is fully differentiable, and the result is a learning framework that is truly end-to-end. We perform extensive experiments on the 3DMatch and Kitti datasets. The experiments demonstrate that our approach benefits significantly from the integration of the learned features and our learning strategy, outperforming the state-of-the-art on Kitti.

Author’s contribution: The author developed the methods, conducted the experiments and was the main contributor to the manuscript.

Paper D: Deep Projective 3D Semantic Segmentation

Felix Järeemo Lawin, Martin Danelljan, Patrik Tosteberg, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. “Deep Projective 3D Semantic Segmentation”. In: *International Conference on Computer Analysis of Images and Patterns (CAIP)*, 2017, pages 95–107. Springer, doi: 10.1007/978-3-319-64689-3_8.

Abstract: Semantic segmentation of 3D point clouds is a challenging problem with numerous real-world applications. While deep learning has revolutionized the field of image semantic segmentation, its impact on point cloud data has been limited so far. Recent attempts, based on 3D deep learning approaches (3D-CNNs), have achieved below-expected results. Such methods require voxelizations of the underlying point cloud data, leading to decreased spatial resolution and increased memory consumption. Additionally, 3D-CNNs greatly suffer from the limited availability of annotated datasets.

In this paper, we propose an alternative framework that avoids the limitations of 3D-CNNs. Instead of directly solving the problem in 3D, we first project the point cloud onto a set of synthetic 2D-images. These images are then used as input to a 2D-CNN, designed for semantic segmentation. Finally, the obtained prediction scores are re-projected to the point cloud to obtain the segmentation results. We further investigate the impact of multiple modalities, such as color, depth and surface normals, in a multi-stream network architecture. Experiments are performed on the recent Semantic3D dataset. Our approach sets a new state-of-the-art by achieving a relative gain of 7.9%, compared to the previous best approach.

Author’s contribution: The idea behind the methods came from the author, Martin Danelljan and Fahad Khan. The author conducted the experiments and was the main contributor to the manuscript.

Paper E: Learning Fast and Robust Target Models for Video Object Segmentation

Andreas Robinson, Felix Järema Lawin, Martin Danelljan, Fahad Shahbaz Khan, and Michael Felsberg. “Learning Fast and Robust Target Models for Video Object Segmentation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 7404-7413. doi: 10.1109/CVPR42600.2020.00743.

Abstract: Video object segmentation (VOS) is a highly challenging problem since the initial mask, defining the target object, is only given at test-time. The main difficulty is to effectively handle appearance changes and similar background objects, while maintaining accurate segmentation. Most previous approaches fine-tune segmentation networks on the first frame, resulting in impractical frame-rates and risk of overfitting. More recent methods integrate generative target appearance models, but either achieve limited robustness or require large amounts of training data.

We propose a novel VOS architecture consisting of two network components. The target appearance model consists of a light-weight module, which is learned during the inference stage using fast optimization techniques to predict a coarse but robust target segmentation. The segmentation model is exclusively trained offline, designed to process the coarse scores into high quality segmentation masks. Our method is fast, easily trainable and remains highly effective in cases of limited training data. We perform extensive experiments on the challenging YouTube-VOS and DAVIS datasets. Our network achieves favorable performance, while operating at higher frame-rates compared to state-of-the-art.

Author’s contribution: The author and Andreas Robinson contributed equally to the writing of the manuscript and the conduction of the experiments. The method was developed in collaboration between the author, Andreas Robinson and Martin Danelljan.

Paper F: Learning What to Learn for Video Object Segmentation

Goutam Bhat, Felix Järema Lawin, Martin Danelljan, Andreas Robinson, Michael Felsberg, Luc Van Gool, and Radu Timofte. “Learning What to Learn for Video Object Segmentation”. In: *16th European Conference on Computer Vision*, 2020, pp. 777-794. Springer, Cham. doi: 10.1007/978-3-030-58536-5_46.

Abstract: Video object segmentation (VOS) is a highly challenging problem, since the target object is only defined by a first-frame reference mask during inference. The problem of how to capture and utilize this limited information to accurately segment the target remains a fundamental research question. We address this by introducing an end-to-end trainable VOS architecture that integrates a differentiable few-shot learner. Our learner is designed to predict a powerful parametric model of the target by minimizing a segmentation error in the first frame. We further go beyond the standard few-shot learning paradigm by learning what our target model should learn in order to maximize segmentation accuracy. We perform extensive experiments on standard benchmarks. Our approach sets a new state-of-the-art on the large-scale YouTube-VOS 2018 dataset by achieving an overall score of 81.5, corresponding to a 2.6% relative improvement over the previous best result.

Author’s contribution: The author and Goutam Bhat contributed equally to the writing of the manuscript and the conduction of experiments. The methods were developed through collaboration between the author and the co-authors.

1.6 Additional Publications

In addition to the included publications of the thesis, the author contributed to the following articles. The first paper is related to the works of this thesis on point set registration. It presents a framework for the evaluation of loss functions used in point set registration algorithms. The second paper is a work on video object segmentation. It extends the method in Paper E to model the target object with channel attention vectors. The proposed method

was presented at the 2019 DAVIS challenge in semi-supervised video object segmentation.

Assessing Losses for Point Set Registration

Anderson CM Tavares, Felix Järemo Lawin, and Per-Erik Forssén. “Assessing Losses for Point Set Registration”. In: *IEEE Robotics and Automation Letters*, 5(2):3360–3367, April 2020, vol. 5, no. 2, pp. 3360-3367, doi: 10.1109/LRA.2020.2976307.

Discriminative Learning and Target Attention for the 2019 DAVIS Challenge on Video Object Segmentation

Andreas Robinson, Felix Järemo Lawin, Martin Danelljan, and Michael Felsberg. “Discriminative Learning and Target Attention for the 2019 DAVIS Challenge on Video Object Segmentation”. In: *The 2019 DAVIS Challenge on Video Object Segmentation. CVPR Workshops*, 2019.

3D perception and registration

This chapter discusses 3D perception and point set registration. First, we investigate sensors for 3D perception and how these measure depth. Specifically, Paper A proposes a method for phase unwrapping of amplitude modulated time-of-flight measurements in depth cameras. Next, this chapter proceeds with the problem of point set registration, which can be used to combine 3D measurements from different camera viewpoints into a single 3D model. In Paper B and Paper C, we propose extensions to probabilistic point set registration methods based on Expectation-Maximization. The final part of the chapter discusses the learning of 3D feature representations for improving registration accuracy.

2.1 Depth cameras

Depth cameras are sensors that generate depth images, i.e. images with pixels that contain measurements of the perceived depth of the observed point in the scene with respect to the camera position. By using the projective geometry of the camera, a depth image can be projected out into a 3D point set. Such point sets can further be used to represent and reconstruct the observed 3D scene. Owing to inventions within sensor chip construction, a large number of depth cameras have been developed in recent years. The most commonly used types of depth cameras in computer vision applications are Lidar scanners and RGB-D cameras. Lidar cameras generally use a spinning scanner that generates a full 360-degree depth image. In contrast, RGB-D cameras have sensor arrays that capture color and depth similar to regular cameras. Figure 2.1 shows examples of depth images and the corresponding 3D point sets for both Lidar and RGB-D cameras.

There exist several different techniques to measure depth. From a calibrated stereo pair of cameras, the depth of corresponding image points can be estimated through triangulation. The main limitation of this technique is

Figure 2.1: Depth images (top) acquired using Ouster Lidar cameras (left) and a Kinect depth camera (right). The Lidar image was acquired at Fisketorget in Västervik, Sweden for the n-dimensional modeling project [58] and a the Kinect depth image is taken from the 3DMatch dataset [126], depicting an indoor scene. The bottom row of images show the corresponding 3D point sets, generated by out projecting the depth images.

that it relies on finding accurate correspondences. This is a hard problem whenever the textures in the image are repetitive or smooth. Consequently, stereo-based depth images are often coarse and contain uncertain measurements.

Depth cameras based on structured light solve the texture problem in stereo systems by emitting fixed patterns of near-infrared light beams [98]. The patterns can then be detected in the camera, and the depth that corresponds to the emitted points can be recovered using triangulation. A downside to this technique is that the patterns must be sufficiently visible for them to be detected, often leading to reduced levels of spatial detail and a low effective range.

A third approach is time-of-flight (ToF), which is used in both Lidar and many RGB-D cameras. Paper A proposes a method for a specific version of ToF based on amplitude modulation. The principle of ToF and the proposed method are further described in the sections below.

2.2 Time-of-flight ranging

In ToF, depth is measured by estimating the time-of-arrival of an emitted light signal. It is performed by emitting a light pulse onto the surrounding scene. The light pulse is then reflected onto the observed surfaces, and eventually some of that light returns to the camera. This is observed as a temporary increase of photons in the sensor chip. Finally, the distance to the observed surface point can be computed as a function of the difference between the time-of-emission and the time-of-arrival Δt ,

$$d = \frac{c\Delta t}{2}. \quad (2.1)$$

Here c is the speed of light. In contrast to stereo and structured light, ToF cameras can resolve higher levels of spatial detail. Further, compared to structured light, the time-of-arrival is detectable at a much larger depth than the emitted fixed pattern, which gives an increased effective range.

There are many challenges with this technique and the sources of errors include shot noise and multi-path reflections. Further, the estimation of the time-of-arrival needs to be extremely accurate to provide reasonably small depth errors. As we can deduce from equation (2.1), an error in the time-of-arrival estimate of just a nanosecond results in a depth error of 15 centimeters. This puts high demands on measurement technology, making depth cameras based on time-of-arrival of a light pulse expensive to manufacture. A more detailed description of the time-of-flight principle can be found in [43].

Amplitude modulated time-of-flight

Instead of tackling the problems of inaccurate time-of-arrival measurements for light pulses, we can alter the shape of the emitted light signal. One popular approach is to perform amplitude modulation of the light [43, 33], i.e. emitting a sinusoidal shaped continuous light wave. Unlike methods based on time-of-arrival, ToF based on amplitude modulation measures the phase shift ϕ of the light wave using correlation with a reference signal. With known modulation frequency f , the depth can be converted into a distance by

$$d = \frac{c\phi}{4\pi f}. \quad (2.2)$$

This technique can yield a more accurate depth estimate than the standard time-of-arrival approach using a light pulse. However, due to the periodicity of the phase measurements, the depth estimates are ambiguous. More precisely, any depths corresponding to phase shifts at different 2π increments, i.e. $\phi + 2\pi n$ where n is an integer, will result in the same measured quantities corresponding to $\phi \in [0, 2\pi)$. Consequently, the unambiguous range for an

amplitude modulated ToF measurement is bounded by $d_{max} = \frac{c}{2f}$. We observe that this range is increased when we reduce the modulation frequency f , but that will simultaneously amplify the noise. For instance, if we assume additive noise on ϕ , the noise on the final depth estimate is proportional to $1/f$.

Another approach for increasing the unambiguous range is to combine multiple depth estimates generated at different modulation frequencies f_i . This gives rise to the following set of equations,

$$\begin{aligned} d &= \frac{c(\phi_0 + 2\pi n_0)}{4\pi f_0} = \frac{c(\phi_1 + 2\pi n_1)}{4\pi f_1} = \dots = \frac{c(\phi_{M-1} + 2\pi n_{M-1})}{4\pi f_{M-1}} \iff \\ \frac{k_0}{2\pi}\phi_0 + k_0 n_0 &= \frac{k_1}{2\pi}\phi_1 + k_1 n_1 = \dots = \frac{k_{M-1}}{2\pi}\phi_{M-1} + k_{M-1} n_{M-1} \end{aligned} \quad (2.3)$$

where the unknowns are the integer coefficients $\{n_i\}_{i=0}^{M-1}$, indicating which interval the phase measurement belongs to. Here we have also simplified the system by setting the constants $\{k_i\}_{i=0}^{M-1}$ to be the least common multiples of $\{f_i\}_{i=0}^{M-1}$ divided by the respective frequency. The process of finding the coefficients n_i is called phase unwrapping.

Phase unwrapping

Since the system of equations in (2.3) is redundant, we cannot find a unique global optimum. Moreover, the discreteness of the problem prevents us from using regular linear solvers. On the other hand, if the fractions f_j/f_i are all rational numbers, the phase measurements will have a common wraparound point at a specific range. Within this range, we can find a local solution to the unwrapping coefficients. This is also illustrated in Figure 2.2, showing the phase measurements corresponding to different depths for the modulation frequencies used in the Kinect v2 camera [1]; 16, 80 and 120 MHz. We further observe that they have a common wraparound at a much larger depth than for the individual frequencies.

A common approach to solve such systems with the above assumptions is to employ the Chinese remainder theorem (CRT) [56, 116]. This method is fast and has been used in drivers for the Kinect v2 camera [1, 118, 80]. However, with the presence of noise, CRT frequently finds the wrong solutions. Other works employ graphical methods combined with constraints on the reflectance [25]. Recent approaches even train deep convnets to directly predict the depth from the raw multi-frequency phase measurements [104]. While such methods can potentially learn to generate accurate depth estimates, they introduce increased computational complexity compared to CRT. In many applications within robotics, the decoding process for the depth camera is a small part of a larger system. Reducing the computational overhead when processing the sensor measurements is therefore desirable. In Paper A, we propose a probabilistic approach for finding the coefficients $\{n_i\}_{i=0}^{M-1}$, which

Figure 2.2: Wrapped phases for tof measurements with modulation frequencies 16, 80 and 120 MHz from top to bottom, in the range 0 to 25 meters. The dashed red line at 18.75 meters marks the common wrap-around point for all three phases. Just before this line we have the intervals corresponding to the unwrapping coefficients 1, 9 and 14.

is both robust and has a speed comparable to CRT. The proposed approach from Paper A is described below.

Kernel density estimate based phase unwrapping

For given maximum range, there exists a finite number of valid solutions to the system in (2.3). Moreover, we can reduce the solution space even further by only considering coefficients $\{n_i\}_{i=0}^{M-1}$ such that they correspond to overlapping depth intervals. We note that, each valid combination of coefficients corresponds to a depth hypothesis. By setting appropriate values for the modulation frequencies [1], the total number of hypotheses will be sufficiently small to compute all of them without significant computational cost.

In Paper A, we investigate such an approach for the Kinect v2 camera. The three modulation frequencies of the Kinect v2 generate a system with 18 valid hypotheses of the unwrapped depth and an ambiguous range of 18.75 meters as depicted in Figure 2.2. A likelihood estimate for these hypotheses being correct can be computed from the residuals derived from the equations

(2.3). In our experiments, we observed that by selecting the hypothesis with the highest likelihood already results in a more robust prediction than the baseline CRT approach.

Next, to improve the robustness of the likelihood estimate, we propose a statistical approach which utilizes the assumption that the true depth image is locally smooth. Specifically, we estimate the distribution of depth hypotheses $p(d)$ over spatial neighborhoods and select the one with the highest likelihood. To this end, we compute a weighted kernel density estimate (KDE) over the hypotheses $i = 0, 1, \dots$ in the neighborhood of each pixel $\mathbf{x}$,

$$p(d^i(\mathbf{x})) = \frac{\sum_{j \in \mathcal{I}, k \in \mathcal{N}(\mathbf{x})} w_{jk} K(d^i(\mathbf{x}) - d^j(\mathbf{x}_k))}{\sum_{j \in \mathcal{I}, k \in \mathcal{N}(\mathbf{x})} w_{jk}}. \quad (2.4)$$

Here $K(\cdot)$ is a Gaussian kernel, and w_{jk} is a sample weight. The sets of samples to be considered are defined by the hypothesis indices $\mathcal{I}$ and by the set of all spatial neighbors $\mathcal{N}(\mathbf{x}) = \{k : \|\mathbf{x}_k - \mathbf{x}\|_1 < r\}$ where r is a square truncation radius. For efficiency, we only consider the top two or three hypotheses in each pixel based on the hypothesis likelihood. The hypothesis weight w_{ik} is computed using a Gaussian kernel in the spatial domain, the hypothesis likelihood based on the residuals of the equations in (2.3) and a phase likelihood based on the amplitude of the received signal (see Paper A for more details).

Finally, we select the depth hypothesis for each pixel that gives the highest kernel density. We further use the estimated likelihood of a depth estimate $p(d^i(\mathbf{x}))$ as a confidence measure. By setting a reasonable threshold on $p(d^i(\mathbf{x}))$, we can suppress outliers that occur in noisy regions and edges of the scene. In the experiments of Paper A, we implemented the approach to decode measurements from the Kinect v2 camera [1]. The experiments demonstrate a significantly increased number of correctly unwrapped depth measurements, indicating that the proposed hypothesis selection is more robust to noise than CRT. This increased robustness is more evident in scenarios with a lot of noise, such as outdoor scenes and image regions with large depths.

2.3 Point set registration

In this section, we delve into the problem of 3D-point set registration. The problem is fundamental to many applications in computer vision, such as 3D mapping and scene understanding. As illustrated in Figure 2.3, the task of point set registration is to align a number of overlapping point sets $\mathcal{X}_i = \{x_{ij}\}_{j=1}^{N_i}$, $i = 1, \dots, M$. Each point set consists of 3D point samples, which can be structured arbitrarily. The different point sets could for instance be the results of acquisitions using cameras at different positions in the scene. By assumption, the point sets are all related by a set of unknown transformations $T_i : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ that map the point sets to a common reference frame of the

Figure 2.3: Point set registration of four point set acquired using a depth camera. We seek to estimate the unknown transformations T_1 to T_4 that aligns point sets to a common reference frame.

scene. Consequently, by finding these transformations, we can align the point sets and construct a complete 3D model of the scene. In this thesis, we focus on point sets that are acquired using 3D cameras, e.g. Lidar or RGB-D cameras (see examples in Figure 2.1). We further assume that the observed scenes are static, leading to rigid registration problems. In other words, $T_i \in SE(3)$ and can be decomposed into a rotation matrix and a translation vector, acting on 3D points as $T_i(x) = R_i x + t_i$.

This thesis focuses on the specific branch of probabilistic methods for registration that finds the parameters of the transformations using the Expectation-Maximization algorithm. The following sections detail methods for registration, with focus on the probabilistic approaches proposed in Paper B and Paper C.

Background and previous works

The majority of methods for point set registration in the literature can be formalized as different approaches to minimize following cost function,

$$L(T_1, \dots, T_M; \mathcal{X}_1, \dots, \mathcal{X}_M) = \sum_{i,j,k,l} \alpha_{ijkl} \rho(T_i(x_{ik}), T_j(x_{jl})). \quad (2.5)$$

Here $\rho \geq 0$ is a function of the distance between points and $\alpha = [\alpha_{ijkl}]$ are correspondence weights, indicating if point k in point set i originates from the same position in the scene as point l in point set j . Indeed, if all point sets are transformed versions of the same point set, all points have exact correspondences and the loss is zero at the global optimum. In a real world scenario,

however, the point sets are noisy and only overlap partially. As a result, the distances between corresponding points are often non-zero and many points do not even have correspondences in the other point sets.

If we restrict to the pair-wise case, the equation can be reduced to,

$$L(T; \mathcal{X}_1, \mathcal{X}_2) = \sum_{k,l} \alpha_{kl} \rho(x_{1k}, T(x_{2l})) \quad (2.6)$$

where we have fixed the position of point set 1, i.e. setting T_1 to equal the identity transformation, using it as a reference model. Consequently, we only need to find a single transformation T that maps the source set to the reference set. With known α , the transformations can easily be found by optimization (2.6) over the parameters of T . If ρ is the L^2 norm, the solution can be found from a closed-form expression derived using the weighted orthogonal Procrustes analysis [50] or DQ Localization [113]. The main challenge arises in the general case when the correspondence weights in α are unknown.

Over the years, many different methods for solving the registration problem have been proposed. The most well-known approach is Iterative Closest Point (ICP) [11, 4, 101]. In these methods, the weights α are set to be binary, indicating for each point in X_1 the nearest point or best match in X_2 , given the current transformation parameters T^n . Next, the cost function is minimized for the L^2 norm in (2.6) over T_2 , and the procedure is repeated until convergence. While the nearest neighbor assignment approach can lead to fast convergence it frequently finds the wrong solution in non-ideal scenarios caused by noise, outliers, and if the initial transformation parameters deviate significantly from the optimal ones.

To overcome these limitations, probabilistic methods use soft assignments in α to allow for partial matches. While α is no longer binary, [39] suggests a corresponding doubly stochastic constraint, i.e. $\sum_j \alpha_{kl} = 1$ $\sum_i \alpha_{kl} = 1$. An initial matrix is first computed using a smooth kernel $g(\cdot, \sigma)$, e.g. Gaussian, where σ controls the smoothness. Next, the constraints can be enforced by normalizing the rows and columns following the Sinkhorn-Knopp algorithm [103]. Finally, the transformation parameters can be computed from closed-form expressions [50, 113]. Note that if $\sigma \rightarrow 0$, Sinkhorn-Knopp converges to a binary assignment matrix, resembling the α in the ICP version of the loss.

An appealing property of probabilistic approaches is that they allow for integrating methods from probability theory and Bayesian statistics. In [15, 47, 82, 49], the registration problem is formulated as a maximum likelihood (ML) optimization procedure over the transformations. As observed in [15], if we only perform a single normalization step of α the probabilistic approach in [39] has a resemblance to Expectation-Maximization (EM), where each point in the reference set is seen as a component in a Gaussian mixture model (GMM). Following this observation, several EM based registration methods have been proposed [47, 82, 49]. In [82], more flexibility is added by includ-

ing σ as a parameter to be optimized. Further generalization was introduced in [28] by including a free GMM parameterized by the means and standard deviations μ, σ . Here, α is defined as indicating correspondences between points to the mixture components. This formulation allows for more flexibility in mixture model construction and enables joint registration of an arbitrary number of point sets. As will be further discussed below, this thesis investigates extensions to the method in [28].

Another branch of methods employs kernel density estimation by setting ρ in (2.5) to a smooth kernel function. To their benefit, losses based on kernel density have improved monotonicity properties [106] compared to the L^2 loss used in ICP. In [108] the kernel is set to an isotropic Gaussian. This was generalized in [73], where the densities are computed using a grid of anisotropic normal distributions. Further generalization was proposed in [53], where the densities of the point sets are first estimated using a GMM. Registration is then performed by minimizing an L^2 distance between densities. Another related approach is proposed in [40], where the registration problem is viewed as an N-body problem and the correspondence weights are derived from a Newtonian gravitation model.

Expectation-maximization for registration

In this section, we will describe the EM framework that is used in many registration approaches, including [82, 28]. EM is a powerful algorithm applicable to many machine learning problems. It has the appealing property that it is guaranteed to converge to a local optimum. Well-written introductions to EM, with descriptions and derivations, can be found in [7, 81].

In EM based registration, we model the point sets $\mathcal{X}_i = \{x_{ij}\}_{j=1}^{N_i}$, $i = 1, \dots, M$ as 3D-point observations $x_{ij} \in \mathbb{R}^3$ independently drawn from the distributions p_{X_i} . We assume that these distributions are all related to a common scene distribution p_V by the rigid 3D transformations $T_{\omega^i} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$, parameterized by ω^i such that $p_V(T_{\omega^i}(x)) = p_{X_i}(x)$. To model p_V , we use a mixture of Gaussian distributions,

$$p_V(v|\theta) = \sum_{k=1}^K \pi_k \mathcal{N}(v; \mu_k, \Sigma_k). \quad (2.7)$$

Here, $\mathcal{N}(v; \mu, \Sigma)$ is a normal density function with mean μ and covariance matrix Σ . Our mixture model is parameterized by the mixture parameters as $\theta = \{\pi_k, \mu_k, \Sigma_k\}_{k=1}^K$. The number of components is denoted by K and π_k is the mixing weight of component k , satisfying $\sum_k \pi_k = 1$. Additionally, a uniform distribution can be added to the GMM to model outliers as in [28]. In EM based registration, we jointly infer $p_V(v|\theta)$, with parameters θ , and the optimal transformation parameters ω^i for $i = 1, \dots, M$ such that the joint likelihood $\prod_i \prod_j p_V(T_{\omega^i}(x_{ij})|\theta)$ is maximized.

The approach in [82], sets the cluster centers to be fixed at the point locations in one of the point sets, i.e $\mu_k = x_{ik}$ and thus K is equal to the number of points in the i th point set. Since the EM iterations involve evaluating the likelihoods $p_V(T_{\omega^i}(x_{ij})|\theta)$ of all points, this leads to high computational complexity for large scale point sets. Further, the arbitrary selection of the model point set can cause biased estimates of the transformation parameters. In contrast, [28] set μ_k to be a free set of parameters. This gives for more flexibility in the optimization of the mixture model, which could result in less sensitivity to outliers [28], especially during registration of multiple point sets jointly. Moreover, it allows for using a fewer number of components, which can have a regularizing effect on the estimates and it reduces the computational complexity.

In the following sections, we simplify the notation by containing all parameters in the model in $\Theta = \{\theta, \omega^1, \dots, \omega^M\}$. For further brevity, we omit the parameters ω^i in the notation for the transformation functions, i.e we set $T_i = T_{\omega^i}$.

Inference

To infer the parameters Θ , we maximize the log-likelihood of the observed points x_{ij} :

$$\mathcal{L}(\Theta; \mathcal{X}_1, \dots, \mathcal{X}_M) = \sum_{i=1}^M \sum_{j=1}^{N_i} \log(p_V(T_i(x_{ij})|\theta)). \quad (2.8)$$

This objective contains a sum over the mixture components in (2.7) within the logarithm, removing the possibility for solutions on closed-form and even makes it hard to optimize using conventional gradient-based methods. The EM algorithm is an elegant way to tackle this issue by introducing the latent assignment variables $Z \in (1, \dots, K)$, assigning points to the mixture components. Given the current parameter state Θ^n , an EM iteration increases the expected complete data log-likelihood objective,

$$\mathcal{Q}(\Theta; \Theta^n) = \sum_{i=1}^M \sum_{j=1}^{N_i} E_{Z|x, \Theta^n} [\log(p_{V,Z}(T_i(x_{ij}), Z|\theta))]. \quad (2.9)$$

which in turn increases the log-likelihood in (2.8). Each iteration is divided into two steps; the E (Expectation) step and the M (Maximization) step. In the E-step, we evaluate the expectation in (2.9), taken over the probability distribution of the assignment variables,

$$\begin{aligned} p_{Z|X_i}(k|x, \Theta) &= \frac{p_{X_i,Z}(x, k|\Theta)}{\sum_{l=1}^K p_{X_i,Z}(x, l|\Theta)} \\ &= \frac{\pi_k \mathcal{N}(T_i(x); \mu_k, \Sigma_k)}{\sum_{l=1}^K \pi_l \mathcal{N}(T_i(x); \mu_l, \Sigma_l)}. \end{aligned} \quad (2.10)$$

This is a simple operation that only requires us to evaluate the densities of our current mixture model at the transformed points. For compactness, we introduce the notation $\alpha_{ijk}^n = p_{Z|X_i}(k|x_{ij}, \Theta^n)$, and note that these correspond to the assignment weights α in (2.5) between the point j in point set i to mixture component k . In the subsequent M-step, we maximize,

$$Q(\Theta; \Theta^n) = \sum_{i=1}^M \sum_{j=1}^{N_i} \sum_{k=1}^K \alpha_{ijk}^n \log(p_{V,Z}(T_i(x_{ij}), k|\theta)) , \quad (2.11)$$

with respect to Θ . The sum over the mixture components is now outside the logarithm, leading to a much simpler objective than the original (2.8). However, due to the non-linearities of (2.11), a closed-form solution to the joint maximization problem over θ and T_i is unlikely to exist. Following [28, 29] and [82], we circumvent this by employing the *expectation-conditional-maximization* (ECM) method. In this approach, the transformations T_i are found first, given the latent assignments and the mixture parameters θ . Next, the mixture parameters are found given the latent assignments and the transformation parameters. By employing isotropic normal distributions in the GMM, i.e. $\Sigma_k = \sigma_k^2 \mathbf{I}$, both sub-problems have closed-form solutions.

Registration is performed by iterating the above EM steps until convergence. Derivations of the closed-form expressions for the updates of T_i , μ_k and Σ_k can be found in [28].

Density variations

Point sets acquired using Lidar cameras generally contain significant density variations. These variations are caused by the functions of the cameras, the geometric structure of the scene and the properties of the observed surfaces. We first note that Lidar point sets follow a spherical sampling pattern, which is usually more dense along the azimuth than the elevation angle. Secondly, the density of the sampling pattern naturally decreases quadratically with the distance from the camera location. Moreover, the further away from the camera, the more likely is the occurrence of missing measurements due to reduced energy in the received signal reflection. As a result, the point sample density significantly decreases in areas far away from the camera. Finally, the local sample density is affected by the direction and the reflective properties of the surface.

Unfortunately, density variations affect the optimization landscape for the loss function in (2.5). This can potentially cause registration approaches to align the dense regions rather than finding the correct registration parameters. Probabilistic EM based approaches suffer gravely from this since the likelihood of the density model will be dominated by the points in the dense regions. In other words, the parameters that give the maximum likelihood of the model may not reveal the true transformation parameters as the locations of the dense regions depend on the camera position. Consequently, in

order to make EM-based approaches useful in practice, we carefully need to consider how to counter the problems of density variations.

If we view this from a probabilistic perspective, the problem induced by density variation can be interpreted as the camera not observing the actual underlying distribution of the scene. Let us define q_V as the underlying distribution and q_{X_i} as the observed distribution of points in point set i . By construction, $q_V(v)$ encodes all 3D-structures, i.e. walls, ground, objects, etc., that is measured by the camera, while q_{X_i} is a result of a distortion of q_V caused by the camera sampling pattern and transformation. Intuitively, it should be easier to register point sets if the points are drawn from q_V rather than q_{X_i} as the distributions then only depend on the transformations.

While the structure of q_V is unknown, a reasonable assumption is that it should be uniform along the surfaces of the scene, and zero outside. The common approach for recovering samples from q_V is to perform downsampling in such a way that the distribution of points along the surfaces becomes uniform throughout the point set. This can be achieved approximately with farthest point sampling [27] or voxel grid re-sampling. As a bi-product, this approach also reduces the computational complexity. On the other hand, by downsampling the point sets, important information about local structures might be lost, potentially reducing registration accuracy. Figure 2.4 shows a point set acquired using a terrestrial Lidar camera that has been downsampled with furthest point sampling at different sampling rates.

Density adaptation

In Paper B we tackle the issues caused by non-uniform point densities by revising the underlying EM formulation and model assumptions. Instead of modeling the density of 3D points, we infer a model of the actual underlying distribution q_V . We model it as a parameterized density function $p_V(v|\theta)$. To this end, we employ a GMM (2.7) and minimize the Kullback-Leibner (KL) divergence from p_V to q_V ,

$$\text{KL}(q_V||p_V) = \int \log \left(\frac{q_V(v)}{p_V(v|\theta)} \right) q_V(v) dv. \quad (2.12)$$

Through the decomposition of the KL-divergence $\text{KL}(q_V||p_V) = H(q_V, p_V) - H(q_V)$ into the cross entropy $H(q_V, p_V)$ and entropy $H(q_V)$ of q_V we can equivalently maximize,

$$\mathcal{E}(\Theta) = -H(q_V, p_V) = \int \log(p_V(v|\theta)) q_V(v) dv \quad (2.13)$$

In (2.13), the integration is performed in the reference frame of the scene. In reality, the 3D points x_{ij} are observed in the coordinate frames of the individual cameras and we relate them with the transformations T_i . By applying

Figure 2.4: Point sets from the VPS dataset [109] downsampled using farthest point sampling [27] with different sampling rates. The top left shows the original point set with no downsampling. As we can see, the point density is significantly higher in the parts of the scene near the camera position. In the top right, 50% of the points are kept, but the density variations are still severe. The sampling rates of the point sets start to look uniform at a rate of 10% or lower (bottom right).

the change of variables $v = T_i x$, we obtain

$$\mathcal{E}(\Theta) = \frac{1}{M} \sum_{i=1}^M \int_{\mathbb{R}^3} \log(p_V(T_i(x)|\theta)) q_V(T_i(x)) |\det(DT_i(x))| dx. \quad (2.14)$$

Here, $|\det(DT_i x)|$ is the absolute value of determinant of the Jacobian of the transformation, which for a rigid transformation equals to 1.

We note that if $\{x_{ij}\}_{i=1}^{N_i}$ are independent samples from $q_V(T_i(x))$, the original ML formulation (2.8) is recovered as a Monte Carlo sampling of the objective (2.14). Therefore, the conventional ML formulation (2.8) relies on the assumption that the observed points x_{ij} follow the underlying uniform distribution of the scene q_V . However, this assumption completely neglects the effects of the acquisition sensor that causes density variations.

In our formulation, we consider the points in set $\mathcal{X}_i$ to be independent samples $x_{ij} \sim q_{X_i}$ of the distribution $q_{X_i}(x)$. In addition to the underlying 3D distribution of the scene q_V , q_{X_i} can also depend on the position and properties of the camera, and the inherent properties of the observed surfaces. This allows more realistic models of the sampling process to be employed. By assuming that the distribution q_V is absolutely continuous [26] with respect to q_{X_i} , equation (2.14) can be written,

$$\mathcal{E}(\Theta) = \sum_{i=1}^M \int_{\mathbb{R}^3} \log(p_V(T_i(x)|\theta)) \frac{q_V(T_i(x))}{q_{X_i}(x)} q_{X_i}(x) dx. \quad (2.15)$$

Here, we have also ignored the factor $1/M$. The fraction $w_i(x) = \frac{q_V(T_i(x))}{q_{X_i}(x)}$ is known as the Radon-Nikodym derivative [26] of the probability distribution $q_V(T_i(x))$ with respect to $q_{X_i}(x)$.

Finally, to compute (2.15), we perform a Monte Carlo approximation as,

$$Q(\Theta, \Theta^n) \approx Q(\Theta, \Theta^n) = \sum_{i=1}^M \frac{1}{N_i} \sum_{j=1}^{N_i} \sum_{k=1}^K \alpha_{ijk}^n w_{ij} \log(p_{V,Z}(T_i(x), k|\theta)). \quad (2.16)$$

Here, for brevity we have defined $w_{ij} = w_i(x_{ij})$. Intuitively, w_{ij} is the ratio between the density in the latent scene distribution and the density of points in point cloud $\mathcal{X}_i$. In Paper B, we introduce two different approximations of $w_i(x)$. One is based on the camera and the observed surfaces, while the other is empirical and only depends on a local density estimate of the points. An example of the empirical weights is provided in Figure 2.5.

Following the above motivation, we modify the EM based registration algorithm in [28] by maximizing the objective in (2.16) in the M-step. The only difference to the original EM formulation is the computations of the point-wise weights w_{ij} , which only needs to be performed once. In the experiments of Paper B, we found that the empirical model improves over both the camera model and re-sampling approaches on several datasets containing Lidar scans with challenging density variations.

Feature based registration

In the above sections, we only considered using the spatial coordinates of the points as information to the registration methods. In this section, we discuss how discriminative features can be employed to improve registration performance. Intuitively, features can provide prior information on the local similarities between regions of point sets when computing the correspondence weights in α in equation (2.5). As an example, ICP has been extended in [37, 79] to combine color features and spatial coordinates in the nearest neighbors search step. Since the nearest neighbor assignment also depends on the similarity and not only closeness in spatial location, these methods achieved increased robustness to initialization.

Figure 2.5: Visualization of the observation weight computed using the empirical method (right) of a point set from the VPS dataset [109] (left). The 3D points in the densely sampled regions in the vicinity of the Lidar position are assigned low weights, while the impact of points in the sparser regions are increased. These illustrations also appear in Paper B.

Another popular approach for feature based registration is to employ the RANSAC framework [32]. In registration methods based on RANSAC [96], a prior correspondence set is first constructed using nearest neighbor assignments in feature space. Next, minimal samples are drawn from the set of point pairs, followed by a registration step and an evaluation step. Finally, the sample that provided the highest evaluation score will be used to provide the final registration parameters.

Features have also been used in probabilistic frameworks. These methods compute a non-binary prior for correspondences from distances in feature space. As in section 2.3, the correspondence weights α can then be refined using Sinkhorn-Knopp [123] or EM [125, 18, 19] in combination with distances in coordinate space.

What features are useful for registration then? Color features, such as raw RGB values or other variants such as HSV, have proven to provide increased robustness in different methods [37, 79, 18]. However, they do not capture local geometric characteristics or contextual information about objects, useful for robustly establishing correspondences for registration. Feature extraction on 3D point sets introduces several challenges due to their unstructured nature. Importantly, 3D point set features for recognition and registration, require to some extent invariance to augmentations of the point sets caused

by e.g. the position of the acquisition sensor. These augmentations include rotations, translations, sampling density, and scale.

Over the years, many different approaches for extracting features have been proposed. Early approaches attempt to capture the local appearance by handcrafting features based on the local geometry. Examples of such features are point feature histograms (PFH) [97, 96] that are based on the relations between surface normals of neighboring points.

The main advantage of handcrafted approaches is that they are applicable without any learning. However, their performance and generalization abilities are limited. The emergence of large-scale datasets of registered point sets [36, 126] has enabled the employment of data-driven methods. This can be observed in more recent approaches, that follow the current trend of learning features by training deep neural networks. These networks are constructed from on architectures based on PointNet [90], voxelized 3D convolutions [78, 13] and variants of graph convolutions[99]. By using annotated correspondences, or correspondences estimated from registered point sets, these networks can be trained using e.g. contrastive learning techniques. The contrastive learning paradigm, employs objective functions such as the triplet loss [100] or its variants [14], that encourages similarity of matches, while reducing similarity of mismatches. This technique of learning features has generated a substantial improvement, both in terms of keypoint matching accuracy [126, 21, 22] and registration robustness in RANSAC based frameworks [122, 22, 14, 20, 67].

Deep features have also been employed in the recent registration methods in [114, 115, 72, 12, 38, 123]. These methods employ a similar approach as the classical approaches described in 2.3, in that they minimize a loss function that resembles (2.5). The main difference is in the computation of the correspondence weights α , where the methods based on deep features perform the matching step in the feature space.

A major challenge in the training step of the aforementioned deep approaches is caused by the n^2 complexity of the matching step. The complexity is further amplified by the high dimensionality of the features and a large number of parameters in the deep networks. In the recent Deep Global Registration method (DGR) [12], the complexity is reduced by only considering the nearest neighbors in the feature space. Unfortunately, nearest neighbor sampling is not differentiable, preventing end-to-end learning of the feature extractor. Consequently, DGR relies on pre-training using e.g. contrastive learning. Contrastive learning is an efficient approach for learning discriminative features, but optimizes the network for the task of feature matching and not for registration. The methods [38, 123] enables differentiability by employing soft assignments. However, in order to keep memory usage within reasonable bounds during training, heavy downsampling or keypoint extraction is required.

In Paper C, we propose a new feature based method, which extends the EM based approach described in the sections 2.3 and 2.3. We further propose a learning strategy, that utilizes end-to-end training of a deep feature extractor in a single phase. It is enabled by the fact that the inference of the probabilistic model is differentiable. Learning is simplified further by employing a low number of mixture components, reducing the computational complexity to be linear with the number of points. Moreover, in contrast to previous deep methods, the formulation retains the ability to jointly register an arbitrary number of point sets during both training and online testing. Finally, instead of computing the weights in (2.16) based on the local densities as in section 2.3, the learning-based strategy allows us to learn a weighting that maximizes registration accuracy.

Probabilistic registration with feature learning

In this section, we describe the method proposed in Paper C for incorporating and learning deep features for EM based point set registration. First, we extend the probabilistic model in sections 2.3 and 2.3 to the joint distribution $(X, Y) \sim p_{X,Y}(x, y)$ over the spatial coordinates x and features y of the points. Specifically, we assume that the features are invariant to the transformations T . Further, as in [18], we also assume that features and spatial locations are independent, conditioned on the assignment variable Z . We then maximize the likelihood for a model over the joint density function $p_{X,Y}(x, y|\Theta, \nu)$. Here, we have introduced feature model parameters ν . The conditional independence between X and Y , and the assumption that Y is invariant to Θ , allow us to factorize p as,

$$\begin{aligned} p(x, y, k|\Theta, \nu) &= p(x, y|k, \Theta, \nu)p(k|\Theta, \nu) \\ &= p(x|k, \Theta)p(k|\Theta, \nu)p(y|k, \nu). \end{aligned} \quad (2.17)$$

Using the above factorization, the E-step turns into evaluation of the following posterior distribution,

$$\begin{aligned} p_{Z|X_i,Y}(k|x, y, \Theta, \nu_k) &= \frac{p_{X_i,Y,Z}(x, y, k|\Theta, \nu_k)}{\sum_{l=1}^K p_{X_i,Y,Z}(x, y, l|\Theta, \nu_l)} \\ &= \frac{p_{X_i|Z}(x|k, \Theta)p_Z(k|\Theta)p_{Y|Z}(y|k, \nu_k)}{\sum_{l=1}^K p_{X_i|Z}(x|l, \Theta)p_Z(l|\Theta)p_{Y|Z}(y|l, \nu_l)} \\ &= \frac{\pi_k \mathcal{N}(T_i(x); \mu_k, \Sigma_k)p_{Y|Z}(y|k, \nu_k)}{\sum_{l=1}^K \pi_l \mathcal{N}(T_i(x); \mu_l, \Sigma_l)p_{Y|Z}(y|l, \nu_l)}. \end{aligned} \quad (2.18)$$

Note that, the terms in the expression are identical to the terms in (2.10), except for the added $p_{Y|Z}(y|k, \nu_k)$. In the M-step, we maximize,

$$\begin{aligned}
 F(\Theta; \Theta^n) &= \\
 &\sum_{i=1}^M \sum_{j=1}^{N_i} \sum_{k=1}^K w_{ij} \alpha_{ijk}^n \log(p_{V,Y,Z}(T_i(x_{ij}), y_{ij}, z = k | \theta, \nu_k)) \\
 &= Q(\Theta; \Theta^n) + \sum_{i=1}^M \sum_{j=1}^{N_i} \sum_{k=1}^K w_{ij} \alpha_{ijk}^n \log p_Y(y_j | z = k, \nu_k). \quad (2.19)
 \end{aligned}$$

Here, Q is identical to the objective in the M-step in equation (2.16) and only depends on T_i and θ . It can thus be maximized using the procedure in section 2.3. The second term contains the feature model and only depends on the parameters ν_k . In total, we optimize the parameters of the original spatial distribution of the probabilistic model along with one feature parameter vector ν_k for each component.

Feature and weight learning

In Paper C, we aim at integrating features predicted by a deep neural network. Further, in contrast to Paper B which computes the weights w_{ij} from local density estimates, we propose in Paper C to learn to predict the weights using the network. To this end, we construct a network that outputs high-dimensional features Y and positive scalar weights w for each point in the point set.

The features Y , provide a discriminative representation of the local appearance of a given point. In order to efficiently model these features and enable end-to-end learning of the network, we propose to use a von Mises-Fisher distribution [75], with parameters ν_k to represent the density $p_Y(y_{ij} | z = k, \nu_k)$. Therefore, we enforce the feature vectors to be normalized, i.e. $\|y\| = 1$. Thus, in the E-step we compute,

$$p_{Y|Z}(y|k, \nu_k) \propto e^{\nu_k^T y / s^2}, \quad (2.20)$$

where s^2 defines the concentration of the distribution. In principle, s^2 can be optimized along with the other parameters of the model, or even learned along with the parameters of the network. In Paper C, however, we fixed s^2 to the same constant value for all mixture components since that yielded stability during training and efficient model updates. In the M-step, we update ν_k by solving

$$\max_{\nu_k} \sum_{i=1}^M \sum_{j=1}^{N_i} w_{ij} \alpha_{ijk}^n \nu_k^T y_{ij}, \quad \text{subject to } \|\nu_k\| = 1. \quad (2.21)$$

The solution to the above maximization problem is

$$\nu_k = \frac{\sum_{i=1}^M \sum_{j=1}^N w_{ij} \alpha_{ijk}^n y_{ij}}{\left\| \sum_{i=1}^M \sum_{j=1}^N w_{ij} \alpha_{ijk}^n y_{ij} \right\|}. \quad (2.22)$$

A derivation of the solution is provided in the supplementary material of Paper C.

Registration loss learning

In Paper C, we learn the parameters of the deep neural network by minimizing a registration loss over a dataset $\mathcal{D} \subset 2^{\{\mathcal{X}_i\}_1^M}$, consisting of D groups $\mathcal{D}_d \in \mathcal{D}$ of overlapping point sets. This loss can be written as

$$\mathcal{L}(\phi; \mathcal{D}) = \sum_{d=1}^D \mathcal{R}(\phi; \mathcal{D}_d), \quad (2.23)$$

where the parameters of the network are contained in ϕ . We denote the number of point sets in each group by $M_d = |\mathcal{D}_d|$.

We further assume that the ground truth relative transformations T_{ik}^{gt} from point set k to i are known in the training set. A sample loss $\mathcal{R}$ is computed over point-wise errors given the estimated relative transformations $T_{ik}^n = (T_i^n)^{-1} \circ T_k^n$ for each EM iteration n as

$$\mathcal{R}(\phi; \mathcal{D}_d) = \sum_n^{N_{\text{iter}}} v_n \sum_{i=1}^{M_d-1} \sum_{k=i+1}^{M_d} \frac{1}{N_i} \sum_{j=1}^{N_i} \rho(\|T_{ik}^n(x_{ij}) - T_{ik}^{\text{gt}}(x_{ij})\|_2 / c), \quad (2.24)$$

where the estimated transformations T_i^n and T_k^n are found using the EM approach described in section 2.3. The loss for each EM iteration is weighted by an increasing sequence of scalars v_n . As a result, each iteration is supervised to move the transformation estimate T_{ik}^n closer to the ground truth T_{ik}^{gt} , but v_n ensures that the final iterations have the highest impact on the loss. To update the parameters ϕ of the network, we back-propagate the gradients of the loss through the EM iterations and the network components.

We observed that large errors in the registration loss can cause instability during training. To counter this we employ a Geman-McClure penalty ρ . This is applied to the Euclidean distance between the points mapped by the estimated transformation and the ground truth transformation. The parameter c is a scale factor that determines the steepness of the error function. Crucially, this robust penalty function ensures that the loss is not dominated by samples far from the solution.

In the experiments of Paper C, we demonstrate that the proposed end-to-end learning strategy, with the loss in (2.24), improves registration performance compared to training the network using contrastive learning. Additionally, we demonstrate that the learning of the weights w_{ij} improved the

Figure 2.6: Visualization of the learned point-wise weighting on pairs of point sets from the 3D Match dataset [126] (top) and the Kitti odometry dataset [36] (bottom). Brighter colors indicate higher weights. These examples also appear in the supplementary material of Paper C.

performance of our approach even further. Interestingly, it turns out that the learned weights do not resemble density adaptation, but rather accentuate specific regions and objects, acting as a soft keypoint extractor (see Figure 2.6).

Segmentation

This chapter presents the works of this thesis within segmentation. Segmentation is a broad topic and involves different types of visual data such as images, videos and 3D point sets. In Paper D, we propose a method for semantic segmentation of large-scale point clouds. Further in Paper E and Paper F, we develop approaches for semi-supervised video object segmentation.

In the proposed methods we construct convnets, which are based on architectures for image segmentation. In order to provide the reader with an introduction to the topic, this chapter begins with a short overview of image segmentation methods.

3.1 Image Segmentation

Image segmentation is a well-studied topic in computer vision. Since the advent of deep learning, the development of methods has been rapid. This development is to a large extent made possible by the construction of large-scale annotated datasets for both image classification [23] and segmentation [69, 30, 128]. The development has been further accelerated by the implementation of fast and efficient frameworks [87, 77, 111] for training convnets, making deep learning easily accessible for the research community.

Most top-performing networks in benchmarks for segmentation follow similar architectural structure. They contain a backbone feature extractor network F_θ and a decoder network D_ϕ (see Figure 3.1), here parameterized by θ and ϕ respectively. The input image $I \in \mathbb{R}^{H \times W \times 3}$ of height H and width W is first processed by $F_\theta : \mathbb{R}^{H \times W \times 3} \rightarrow \mathbb{R}^{H^l \times W^l \times C^l}, l = 1, \dots, L$, which outputs L feature maps $\{x^l\}_{l=1}^L = F_\theta(I)$ with a high number of channels C^l (recall the illustration in Figure 1.1). These features are then transformed by $D_\phi : \mathbb{R}^{H^1 \times W^1 \times C^1} \times \dots \times \mathbb{R}^{H^L \times W^L \times C^L} \rightarrow \mathbb{R}^{H \times W \times C}$ into a segmentation map $\tilde{y} = D_\phi(\{x^l\}_{l=1}^L)$ that indicates the predicted probability for all classes

Figure 3.1: Illustration of a Unet based segmentation network containing a feature extractor and a decoder network, both consisting of four conv-blocks. Each block in the decoder network combines features from a corresponding block in the feature extractor with the output of the previous decoder block and passes it to the next. The final decoder block predicts a segmentation map with the same resolution as the input image.

$c = 1, \dots, C$ in each pixel in the image. Given a dataset of images with per-pixel class annotations, the network parameters θ and ϕ can be learned by minimizing a segmentation loss function, based on e.g. pixel-wise cross entropy or Lovasz-Softmax loss [3].

Typically, the backbone feature extractor is constructed by adopting an image classification network such as ResNet [46] or VGG [102]. These networks contain a large number of convolutional layers that produce image features. The ResNet architecture is usually divided conceptually into four blocks of convolutional layers with residual connections. To increase the aggregation of feature information from neighborhood regions of pixels, as well as reducing the memory consumption, each block also performs downsampling through either a pooling or a strided convolution layer. The final layers of the ResNet and VGG architectures are fully connected and designed to produce an image classification prediction.

In segmentation networks, the final fully connected layers of the backbone are generally removed since they lack the spatial information needed for pixel-wise predictions. In contrast, the earlier layers retain features that are useful for the segmentation task. The shallow layers produce features containing local details, object edges and primitive shapes, while the deeper layers produce features with semantic information about objects and structures (recall Figure 1.1). These properties of the feature extractors can be learned during training for image classification [71]. Since ground truth annotations for segmentation are labor-intensive to generate, many methods rely on pre-training for image classification where there already exists an abundance of annotated data [23]. The pre-trained feature extractors can then be reused, or

transferred to the image segmentation task, potentially leading to a reduced demand for diverse annotated segmentation datasets.

The decoder network D_ϕ is assigned the task of mapping the encoded information in the deep features from the backbone into a segmentation map with the same resolution as the input image. Numerous architectures for D_ϕ have been proposed over the years. In essence, these architectures are designed to combine features of different levels of detail to utilize both local structures and contextual semantic information when performing the pixel-wise class predictions.

Fully Convolutional Networks (FCN) [71] demonstrate that fairly accurate segmentation can be achieved using quite elementary decoders. As a first attempt, FCN applies bilinear upsampling of the deepest feature map before the fully connected layers, followed by a projection layer that maps the features to class predictions. This simple approach resulted in a remarkable segmentation accuracy for its time, despite that the deep feature map had a down-sampling factor of 32 compared to the input image. To improve the segmentation accuracy, FCN also suggests a Unet structure with skip connections to sequentially decode features from the deep layer blocks and combine them with features from the shallower blocks (see illustration in Figure 3.1). In the Unet structure, each layer block of the backbone is assigned a decoder block. Each of these decoder blocks upsamples and fuses features from the backbone with outputs from a decoder block corresponding to a deeper backbone layer. The last block performs a final upsampling step and outputs the segmentation prediction. As a result, Unets can exploit both the high-level semantic information in the low-resolution deep features and the spatial details encoded in the shallower features. In FCN, the fusion in the decoder blocks consists of simple concatenation layers. More recent approaches propose more refined fusion techniques. Examples of these techniques are channel attention layers [124], multi-scale refinement layers [68] and a combination of both [34].

A different approach for maintaining high-resolution information in the features is to employ atrous convolutions [2], also referred to as dilated convolutions. Atrous convolutions expand the filter kernels in the convolutional operators with zeros. This way, the receptive field of the layers is increased, enabling similar aggregation of information as generated by pooling and strided convolutions, but without reducing the resolution of the feature maps. Approaches such as DeepLab [10] and PSPNet [127] employ atrous convolutions and achieve impressive segmentation accuracy without skip connections between layer blocks and Unet structures. On the downside, due to the higher resolutions in the backbone features, the computational complexity and memory consumption during training of these networks are higher than Unet based approaches.

3.2 3D Point Set Semantic Segmentation

The development of 3D acquisition sensors such as Lidar and RGB-D cameras has enabled dense reconstruction of large-scale 3D scenes. As these point sets provides accurate representations of the scenes, we are interested in further analyzing them to extract high-level information for scene understanding. In addition to being interesting, 3D scene understanding has many applications within robotics and autonomous driving. One of the major topics within 3D scene understanding is semantic segmentation.

In order to achieve accurate segmentation of 3D point sets, many challenges need to be addressed. In contrast to the regular grid structure of image data, point sets are in general sparse and unstructured, preventing us from directly applying regular CNN architectures. A common strategy is to resort to voxelization in order to apply convnets in 3D [78]. However, this introduces a radical increase in memory consumption and leads to a decrease in spatial resolution. More recent methods improve the efficiency of voxelized 3D convolutions by exploiting the sparsity of the point sets in 3D [13]. Another branch of deep network approaches employ pointnet architectures [90] or its variants [91, 70]. These can directly be applied to unordered 3D points using multilayer perceptrons and different local aggregation operators [70]. A downside to all these methods is their reliance on labeled 3D data, which is scarce due to difficulties in data annotation.

In Paper D, we investigate an alternative approach that avoids the difficulties induced by deep 3D networks. We propose a framework for 3D semantic segmentation that exploits the advantages of deep image segmentation approaches. In this framework, the point sets are first projected onto a set of synthetic images corresponding to virtual camera views. These images are then used as input to a convnet for image segmentation. The resulting pixel-wise segmentation scores are re-projected into the point set and the semantic labels of points are obtained by fusing scores over the different views. Further, we investigate the impact of different input modalities, such as color, depth, and surface normals, extracted from the point set. These modalities are fused in a multi-stream network architecture to obtain the prediction scores in the images. An overview of the method is illustrated in Figure 3.2.

Compared to semantic segmentation methods based on networks applied directly on 3D data, this approach has some major advantages. Since our network operates on images, it can benefit from the abundance of the already existing data sets for image segmentation and classification, such as ImageNet [23] and ADE20K [128]. This significantly reduces or even eliminates the need for 3D data for training purposes. Further, by avoiding the large memory complexity induced by voxelization [78], our method achieves a higher spatial resolution which enables improved segmentation quality.

Figure 3.2: Illustration of the point set segmentation method in Paper D (this figure also appears in Paper D). The input point set is projected into multiple virtual camera views, generating 2D color, depth and surface normal images. The images for each view are processed by a multi-stream CNN for semantic segmentation. The output prediction scores from all views are fused into a single prediction for each point, resulting in a 3D semantic segmentation of the point set.

Point set rendering

In Paper D, we aim to render photo-realistic images from the point sets. This further allows us to exploit convnets that can be pre-trained on image datasets, leading to less dependency on annotated 3D datasets. A common approach to achieve high-quality rendering of 3D objects is to render them as textured meshes. However, for large-scale unstructured point sets, that would require us to perform surface reconstruction. That is a challenging

problem on its own when the point sets contain noise, density variations and regions with missing points.

In our method, we instead render the point sets into synthetic images with a splatting technique [105, 84]. Splatting can be processed in image space, enabling efficient and easily parallelizable implementations. It is performed by projecting the points into the image planes of virtual camera views. Since the point sets are sparse in many regions, simply mapping them to the nearest pixel locations could result in gaps in the rendered images. Such gaps in the rendered images reduces realism and therefore also compromises the performance of image segmentation networks. To tackle this issue, we assign a spatial spread to the projected points, modeled with a truncated normal distribution centered at the projection locations. By selecting an appropriate variance to the spread function, we ensure that the points overlap with their neighbors. As a result, we can generate dense images while avoiding over-smoothing. To handle multi-modal pixel values caused by the overlapping points, we first perform mean shift clustering along the depth direction for each pixel. We then select a cluster to represent the output pixel value based on a foreground prior and the kernel density value, see Paper D for the full details. An illustration of the rendering with examples is provided in Figure 3.3.

During inference, we must cover all points to ensure that they are classified, i.e. all points must be visible in at least one virtual view. To this end, we collect images by rotating the virtual camera 360° around a fixed vertical axis at four different pitch angles and further enforce the images of neighboring virtual cameras to overlap. Such a viewpoint selection approach was sufficient for covering all points in the point sets of the Semantic3D benchmark [42] since they were collected using a standing terrestrial Lidar sensor. In the more general setting, when the point sets can have any structure, the selection of viewpoints needs to be carefully considered. Other recent approaches employ random sampling techniques [8, 59] with the rationale that a large enough sample set will almost surely give full coverage. On the other hand, the selected camera views might not be optimal for segmentation in the sense that the generated images will not provide a realistic appearance. This might result in false predictions generated by the segmentation network. The problem of selecting viewpoints in a manner that optimizes segmentation accuracy is still unsolved.

Multi-stream segmentation network

By employing the procedure in section 3.2, we can render dense color and depth images from the point set. These images can then be processed by a convnet for image-based semantic segmentation. In our approach we employ a three-stream network, taking the RGB image, the depth image converted into a jet color map, and an image containing normal vectors estimated from the image gradients of the depth image. The streams are fused with late fu-

Figure 3.3: Illustration of the splatting based rendering of a point set from the Semantic3D dataset [42]. First, a viewpoint of a synthetic camera is selected to capture a section of the point set (top left). The points are then projected and splatted onto the pixel array of the camera plane (top right). The splatted points are assigned a spatial spread such that they overlap several pixels in the image, here visualized as green circles. To handle multi-modal pixel values, the points are clustered with respect to the depth. Finally, we select one cluster to represent the pixel color (bottom left) and depth (bottom right).

sion, i.e. the output segmentation scores of the streams are simply added together. As we demonstrate in Paper D, the different streams contain complementary information and the best performance is obtained when all streams are combined in the prediction. The rendered images can be mapped to segmentation scores by applying the multi-stream convnet. For each 3D point, the class is selected from the aggregated segmentation scores from all images where the point is visible.

Each stream of the network is processed using a separate FCN [71] model with a VGG16 backbone network [102]. We note that any convnet architecture for 2D semantic segmentation can be employed in our framework. The convnets are trained on a dataset constructed by projecting point sets from

Figure 3.4: Examples of VOS predictions generated by the approach in Paper F. The left-most image is the reference frame (frame 0) with the given mask annotations of the targets overlayed. In this example, the targets are a skater and his skateboard. The VOS network predicts the segmentation masks of the targets in the test frames. These are shown for frames 10, 32 and 72, in images 2-4 from the left.

the Semantic3D [42] train split. To generate segmentation supervision, we render the labels of the points using the method in section 3.2. We then train our full multi-stream network on the rendered image dataset by minimizing the pixel-wise cross entropy loss between the predicted class probabilities and the rendered label images. Finally, we note that in our approach we can benefit from pre-training on large-scale image datasets. Paper D, initializes the weights of the networks for each stream by performing pre-training on ImageNet [23]. In future works, it would be interesting to additionally pre-train the networks on image segmentation datasets [128] to obtain initial network parameters that are more suitable for segmentation.

3.3 Video object segmentation

Image segmentation naturally extends to video segmentation, where the input data is a video sequence and the desired output is a pixel-wise classification of all frames of the video. In this thesis, we investigate the topic of semi-supervised video object segmentation (VOS). In VOS, the assumption is that we are given the masks of target objects in the initial frame. Such target masks can for instance be obtained through manual annotation or by the predictions from an object instance segmentation network [44]. The task is then to automatically track and mark the occupied pixels of the targets objects in all images of the video sequence. An example of how VOS can be successfully performed with the approach proposed in Paper F is shown in Figure 3.4.

The topic of VOS has seen an increased industrial interest in recent years due to its many applications within autonomous driving, robotics, surveillance and video editing. Additionally, there has been a rapid development of methods, much owing to the construction and annotation of VOS datasets such as YouTube-VOS [119] and DAVIS [88].

Background and previous works

Let $I_t \in \mathbb{R}^{H \times W \times 3}$, $t = 0, 1, \dots$ be image frames of a video sequence. We assume that the video captures an object that occupies a subset of pixels that can be marked by a segmentation mask y_t in each frame of the video. In semi-supervised VOS, we are given y_0 , i.e. the mask annotation in the first frame or the *reference frame*. The task is to find the corresponding objects masks y_t in all subsequent frames of the video, which we denote the *test frames*. In the following sections, we focus on the causal case, i.e. the predictions of y_t should only depend on the current and previously observed frames $t = 0, \dots, i$. The non-causal setting also has interesting applications whenever the processing can be handled offline. However, it cannot be applied in an online setting where future frames are not observable.

As the task description indicates, a VOS method is required to find the segmentation masks y_t in the test frames, given only the target mask y_0 in the reference frame. Notably, the category of the target object is unknown. Thus we cannot simply rely on pre-trained object detectors and segmentation networks to perform VOS through re-detection. Instead, we must learn how to adapt to the current target in the video sequence. We note that VOS is a few-shot learning problem and therefore shares many aspects with methods for few-shot classification [31, 66] and segmentation [24].

All top-performing methods for VOS base their models on network architectures for semantic segmentation, such as those described in section 3.1. The main challenge in VOS is to adapt the network to the current target given only the initial frame information. In recent years, many different methods have been proposed.

Earlier approaches [9, 51, 74, 89, 119] used pre-trained segmentation networks and adapted them to the current sequence by performing online fine-tuning on the initial frame. This approach has some major disadvantages. Firstly, adapting the full deep network with all its parameters to a single training example leads to severe overfitting. Secondly, the re-optimization is extremely slow since it requires iterative optimization of the deep network using back-propagation. Finally, the online finetuning approach does not allow offline training to learn representations adapted for video object segmentation since all parameters are re-optimized for each video example. Instead, these methods rely on pre-training on image segmentation.

To remedy these issues, more recent methods integrate target-specific appearance models into the segmentation network. While the target model handles the adaptation and conditioning, the segmentation network can be trained offline and kept fixed during inference. This allows for much faster inference since the adaptation of target models is more lightweight than optimizing deep networks. Moreover, unlike the finetuning-based approaches, these methods avoid overfitting the segmentation networks to the initial tar-

get appearance. They can instead train them to learn generic VOS capabilities.

The most common approach in prior works for constructing target models is to embed the reference target mask with image features into a target representation. The representation is then used to re-detect the target in the incoming frames. In RGMP [85], the representation acts as a spatial attention map and is simply concatenated with the current-frame features, before being input to the segmentation decoder. Similarly, OSNM [120] predicts an attention vector, which acts on the incoming features as a channel-wise modulation. In [54], a generative model is learned from the embedded features corresponding to the initial target mask. The generative model is then used to classify features from the incoming frames. Other works [52, 112], directly store foreground features and classify pixels in the incoming frames through feature matching. Similarly, the method in [117] predicts template correlation filters given the input target mask. Target classification in the test frame is then performed by applying the correlation filters. The feature matching approach in [52, 112] was further extended in [121], by including background features and target attention modulation vectors into the target model. Finally, the STM approach [86] performs feature matching within a space-time memory network. Similar to transformer networks [110], STM performs a read operation, which retrieves information from the encoded memory through an attention mechanism. This information is then sent to the segmentation decoder to predict the target mask.

The above methods based on feature matching and target mask embeddings have shown impressive performance. They constitute a simple but effective way of both modeling the target and subsequently conditioning the segmentation network. Importantly, these target models are differentiable, enabling the training of the full architecture in an end-to-end manner. A drawback to the approaches is the difficulty to learn rich embeddings that generalize to unseen objects. Consequently, these methods generally require a large amount of training data.

As target mask annotations in videos are expensive to create, many approaches rely on synthetically generated video sequences using rendering techniques [57] and image warping methods [85, 117]. While most methods in literature perform comparisons on publicly available VOS datasets such as DAVIS and YouTube-VOS, many approaches also utilize pre-training on a variety of additional training sets, including the mentioned synthetic procedures. For this reason, comparisons between different methods are somewhat problematic.

In Paper E and Paper F, we use a different approach for constructing a target model within a VOS architecture. Similar to visual trackers based on discriminative filter [17, 16, 5], our target model composes a convolutional operator. In our approach, for each target, we optimize an objective based on the target mask annotation in the first frame to learn the target model pa-

rameters. The target model is then applied to incoming image features to predict a target representation that is subsequently refined into a segmentation mask. Our approach is fundamentally different from methods based on on-line finetuning, since our parametric model is lightweight and integrated into a segmentation network, which has fixed weights during inference. Therefore, we can efficiently optimize the parameters of the target model for each target and integrate this procedure in the training of the segmentation network. Furthermore, our approach provides greater adaptivity and generalizability to unseen objects and scenes compared to target models based on feature embeddings and matching. This is further shown experimentally as our approach achieves impressive performance without the need for additional pre-training on segmentation data and synthesized videos.

VOS with optimization-based target models

This section details the VOS methods introduced in Paper E and Paper F. Paper E can be regarded as a predecessor to Paper F since it was first to propose the optimization-based target model. The methods share similar architectures and differ mainly in the learning of the target model and how it is incorporated into the training of the VOS network. Therefore, the methods are described here within the same framework.

Let us denote $S_{\theta, \phi, \tau}$ as our VOS network. It is composed of three modules; a backbone feature extractor F , a decoder network D and a target model T_τ . As in section 3.1, θ and ϕ contains the parameters of F and D respectively. In our approach, these are kept fixed during inference, while T_τ is renewed for each target object in every video sequence. Similar to architectures for image segmentation (see section 3.1) F extracts features which are decoded by D to output a target segmentation map, but here T_τ conditions the features that are input to D with a predicted intermediate representation of the target. The conditioning guides our VOS network to segment the specific target objects that are annotated in the reference frame. Our objective is to learn the parameters θ and ϕ , such that the VOS network can accurately segment the test frames using the target conditioning that is instantiated by the target model.

Intuitively, the parameters θ and ϕ are learned for target agnostic segmentation and generic VOS properties, while τ encodes the target-specific appearance. If we compare this framework to the feature matching approaches in [121, 52, 112, 86], the target state τ corresponds to the feature embedding generated from the previous frames and masks. Contrary to these approaches, our target model T_τ is composed by a parametric operator, designed to predict a target score representation $s = T_\tau(x)$ given the image features $x = F(I)$ from the backbone network.

In order to find the parameters τ , we minimize a supervised objective

$$\tau = \arg \min_{\tau'} L(T_{\tau'}(x), E(y)), \quad (3.1)$$

where $E : \mathbb{R}^{H \times W \times 1} \rightarrow \mathbb{R}^{H^T \times W^T \times C^T}$ is a label function applied to the target mask y to generate the labels of the objective L . Here $H^T \times W^T$ and C^T are the spatial resolution and number of channels of the labels respectively. The simplest version of E is the identity map, which corresponds to the labels the objective being equal to the target masks. However, other more generic representations could also be considered as we discuss later on.

Inference

During inference, we first optimize the target model parameters by minimizing the objective (3.1) given the reference frame image features x_0 and the labels generated by E from the target mask annotation y_0 . In the subsequent test frames $t = 1, 2, \dots$, we extract the features x_t and apply the target model to predict the target score map s_t . The features and scores, are then input to the decoder network D , which outputs the final segmentation mask prediction $\tilde{y}_t = D(x_t, s_t)$. Figure 3.5 illustrates the architecture as it is deployed during inference.

The advantage of the proposed optimization-based target model is that the parameters τ are found by directly minimizing the target prediction error in the first frame. This ensures robust target predictions in the incoming frames since consecutive video frames are highly correlated. Contrary to feature embedding and matching-based approaches, the proposed target model can be optimized to predict target representations without learning highly descriptive and VOS specific feature embeddings.

Notice that our target model and prediction of the target mask are strictly based on appearance. In practice, the described VOS procedure can also benefit from the usage of spatiotemporal information such as prior knowledge about the target position in the image at any time instance or even motion models. In Paper F we use the previous prediction of the target to estimate the target location in the current frame. We then crop the image to limit the search space for possible locations of target prediction. However, such assumptions about the target position must be carefully considered since the target movements and occupied regions occasionally change rapidly and unpredictably between frames. As an example, if parts of the object have previously been occluded, they might reappear in the current frame far from the previous target mask regions in the image (far in the sense of distance in image pixel coordinates). By cropping the image, the reappearing regions of the target might be excluded, ultimately resulting in false negatives in the mask predictions.

Training

The network $S_{\theta, \phi, \tau}$ can be trained using annotated VOS data. For a given sample consisting of a reference frame (I_0, y_0) and a number of test frames $I_t, t =$

Figure 3.5: Illustration of our VOS network. First, we optimize the target model parameters τ by minimizing an error given extracted features x_0 from the reference frame image I_0 and the generated labels $E(y_0)$ from the reference frame mask annotation y_0 . In the incoming test frames I_t , we extract the features x_t and apply the target model to predict a target representation s_t . Lastly, the decoder takes the prediction and backbone features and outputs the final segmentation mask $\tilde{y}_t$ for the test frame.

1, 2, ..., we can perform VOS according to the process described previously in section 3.3, and compute a segmentation loss using ground truth segmentation masks. We can then back-propagate the gradients of the loss through the computational graph of the VOS procedure and update the weights θ and ϕ of F and D . However, this includes passing the gradients through the optimization of the target model parameters τ , requiring the optimization procedure to be differentiable, computationally efficient, and not too memory demanding.

Initially, in Paper E, we only train the decoder network and keep the weights θ of the pre-trained feature extractor fixed. This approach allows us to employ any optimization technique for the target model that ensures accurate target model predictions without exceeding memory constraints during training. Utilizing a ResNet backbone feature extractor pre-trained on ImageNet, the method achieves competitive performance at comparably high

Figure 3.6: Visualizations of the outputs produced by the method in Paper E on the 48th frame of the dance-twirl sequence in the DAVIS dataset [88]. The left image shows the final segmentation predicted by the VOS network, overlaid on the input image. The right image visualizes the output segmentation scores predicted by the target model. Despite being linear, this intermediate prediction is surprisingly accurate. We further observe that the final refined prediction accurately segments the dancer while all the noise in the background produced by the target model is suppressed..

speed. Subsequently, in Paper F, we demonstrate the advantage of training the entire network end-to-end. We show that, by learning the parameters of the backbone feature extractor jointly with the other components of the network during VOS training, we outperform the method in Paper E. In the following sections, we describe the target model and online optimization in more detail.

Target model

The design of the target model, objective function, and the choice of optimization strategy are essential to the performance of our approach. Crucially, the target model should be efficient to enable fast processing, and it should be robust to changes in target and the background appearance. While a complex model, such as a deep neural network, can be optimized into a powerful classifier, it may also be prone to overfitting and computationally inefficient. For that reason, we investigate more lightweight models that rely on discriminative deep features.

In Paper E, we observed that by utilizing deep features from a pre-trained ResNet feature extractor F , the foreground pixels can be coarsely separated from the background using a linear classifier (see the example in Figure 3.6). More precisely, the target model T_τ in Paper E is a composition of two convolutional layers, where the parameters $\tau = (\tau_0, \tau_1)$ encodes the filter weights. Given an incoming feature map x , we apply the filters to obtain a target score

map $s = T_\tau(x) = \tau_1 * (\tau_0 * x)$. The first layer τ_0 , is implemented as a 1×1 convolutional filter that projects high-dimensional backbone features into an intermediate feature map with a lower number of channels. These features are then mapped by the 3×3 conv-layer τ_1 to the single-channel output scores s . While leveraging the same contextual information as a regular 3×3 conv-filter, this composition reduces the total number of parameters, which in turn reduces the risk of overfitting to the initial target appearance and makes optimization more efficient.

In Paper F, we instead use a single 3×3 conv-filter $s = T_\tau(x) = \tau * x$. In order to reduce the dimensionality of the input features x we learn a projection layer during offline training. This layer projects features to a lower number of channels before they are processed by the target model, reducing the required number of parameters in τ . By allowing back-propagation of gradients through the optimization of the target model during training, this projection layer can be learned jointly with the other network components. As a result, we can learn to predict features that are more customized to the target model.

To find the parameters τ we minimize a weighted squared error between the target model outputs $T_\tau(x)$ and the labels $E(y_t)$

$$L(\tau) = \frac{1}{2} \sum_{(x_t, y_t) \in D} \gamma_t \|W(y_t) \cdot (T_\tau(x_t) - E(y_t))\|^2 + \frac{\lambda}{2} \|\tau\|^2. \quad (3.2)$$

Here, $\lambda > 0$ is a hyper-parameter that scales the L^2 -regularization. The function $W : \mathbb{R}^{H \times W \times 1} \rightarrow \mathbb{R}^{\tilde{H}^T \times W^T \times C^T}$ outputs weights that can be used to increase the importance of certain elements of the loss, e.g. balancing the ratio between foreground and background regions. The learning of the target model parameters is performed over a dataset D of feature-label pairs $(x_t, y_t) \in D$, and each sample is weighted by γ_t . During inference, the target model parameters are first assigned initial values, e.g. zeros. The initial dataset only contains the reference frame features x_0 and the given mask annotation y_0 . To provide the dataset with more variability, Paper E creates additional samples by performing data augmentation of the reference frame. The target model is then optimized to minimize the loss in (3.2) over the dataset. In the subsequent test frames, the VOS procedure continues as described above in section 3.3, employing the learned target model and the decoder network on the incoming features to produce the mask predictions $\tilde{y}_t$.

An advantage to our optimization-based formulation is that it allows us to continually update our target model during inference to adapt it to appearance changes in the video sequence. To achieve this we utilize the network predictions in the test frames. Before we receive the next frame $t + 1$, we extend the dataset with the predicted segmentation mask $\tilde{y}_t$ and features x_t . We then re-optimize the target model parameters over this extended dataset. The whole procedure is repeated until the end of the sequence. By setting the

weights γ_t with exponential decay, the impact of older frames is reduced over time. As a result, the target model can adapt to the changes in the appearance of the target and background. However, if the network mask predictions are erroneous, this approach can cause drifting, e.g. the target model will learn to give false predictions and we might lose track of the target. To remedy this potential problem, we always make sure to keep a significant weight γ_0 to the first frame sample in the dataset.

Optimization

There exist numerous efficient solvers for general non-linear quadratic problems of the type in (3.2). In Paper E, we employ a Gauss-Newton (GN) based strategy, which generally has faster convergence properties than standard gradient descent-based methods [76]. In GN, the objective is minimized by iterating between performing quadratic approximations given the current parameter estimate and minimizing approximated objective. Specifically, for each GN iteration, the optimal increment $\Delta\tau$ is found by minimizing the following approximation of the loss in (3.2)

$$\mathcal{L}_D(\tau + \Delta\tau) \approx \Delta\tau^T J_\tau^T J_\tau \Delta\tau + 2\Delta\tau^T J_\tau^T r_\tau + r_\tau^T r_\tau. \quad (3.3)$$

Here, r_τ concatenates all residuals $\sqrt{\gamma_t}W(y_t) \cdot (T_\tau(x_t) - E(y_t))$ and $\sqrt{\lambda}\tau$, and J_τ is the Jacobian of the residuals r_τ at τ . The objective (3.3) is an unconstrained positive-definite quadratic problem, which we minimize over $\Delta\tau$. We then update $\tau \leftarrow \tau + \Delta\tau$ and execute the next GN iteration.

Note that since the target model in Paper F only contains a single conv-filter, the corresponding objective can directly be written in the form of (3.3) without approximations.

Next, we need an efficient strategy for minimizing the positive-definite quadratic problem. First, we note that such problems have solutions that can be expressed in closed-form. However, due to the high number of parameters, such approaches require extensive matrix multiplications and solutions to large-scale linear systems, making them impractical for real-time VOS. Moreover, closed-form solvers cannot utilize the convolutional structure of the problem. In our approach, we instead employ iterative solvers that find approximate solutions.

In Paper E, we use the conjugate gradient descent method [48] and in Paper F we use a simpler steepest descent approach (see the appendix of Paper F for a detailed derivation of how we compute the steepest descent steps). Both methods find the learning rates for the optimization steps adaptively, providing solutions adequately close to the global optimum in few iterations. Furthermore, both conjugate gradient descent and steepest descent steps are differentiable, enabling back-propagation of gradients through the optimizer during training. As a result, we can learn feature representations that are bet-

ter suited for the target model learning, further reducing the need for taking a large number of descent steps.

We note that a low number of optimization steps is desirable since it reduces the computational complexity and gives faster run times. Since the output of the target model is only used to condition the decoder network, it does not need to be optimal with respect to the objective. We observe that the decoder learns to predict accurate segmentation masks although the target model generates uncertain predictions. During network training, it may even be beneficial to provide the decoder with uncertain target model predictions since it can induce robustness to noise during inference. For instance, in Paper F, we only use 5 iterations during training and 20 during inference on the first frame. During the between frame updates of the target model described above in section 3.3, we can initialize the parameters of the target model with the previous ones. Since the changes between consecutive frames are often small, the optimization only requires a few steps to converge. Consequently, our optimization-based methods can operate during inference at comparably high frame rates.

Learning What to Learn

In equation (3.1), we introduced the function E that generates the labels used in the objective. These labels define how the network encodes the target, and it is the representation that the target model predicts to condition the decoder network when it performs target segmentation.

In Paper E, we set the target mask y itself as the label, i.e. E is the identity map. This results in a target model that is optimized to predict the target mask from the input features. The task of the decoder network is then to refine this prediction into a high-quality segmentation mask. While this approach performs competitively, a single-channel mask is unlikely to be the optimal representation of the target within the network. Other representations of the target could provide improved guidance to the decoder network. One could for instance consider cues such as edges, background information, and target key points.

In Paper F, we propose to model E as a network module. We design it to take the target mask as input and output a multi-channel label encoding. We further propose to learn the parameters of E jointly with the other network components of the VOS architecture during training. In other words, we learn what the target model should learn. With this approach, we could in theory generate a richer target representation than only using the target mask. The representation can further provide the decoder network with useful information for target segmentation that is not present in the basic target mask prediction. In addition, we also learn the weighting function W in (3.2), which provides an adaptive attention map to the elements in the loss. Since the network is trained in an end-to-end manner, with supervision on the out-

Figure 3.7: Visualization of learned mask encodings (the bottom three examples also appear in the appendix of Paper F). The left-most column contains the ground truth masks with the corresponding image below. In the other columns, we show labels generated by the label generator in the top rows and absolute values of the corresponding importance weights in the rows below.

put mask predictions from the decoder network, we can learn both a label generator E and a weighting W that maximizes the segmentation accuracy.

In our experiments on VOS benchmarks in Paper F, we demonstrate that our learning what to learn strategy outperforms the basic identity mapping of the target mask as target representation. We further demonstrate improved performance compared to all previous methods based on embeddings, feature matching and online finetuning. Figure 3.7 shows a selection of channels of the label encoding predicted by E and corresponding predicted weights W . We see that the different channels of the label encoding model different aspects of the target, such as background, boundaries and low-resolution versions of the original target mask.

The idea of learning the labels in a supervised objective is generic, in that it can be applied in other few-shot learning frameworks. This has already been demonstrated in works for image correspondence detection [107] and few-shot segmentation [55].

Decoder network

The task of the decoder network D is to predict the final target segmentation mask using the information contained in the target model prediction s and the

backbone features x . As the target model is applied to deep features, its predictions are spatially coarse. We must therefore utilize the spatial information contained in the backbone features to recover a high-quality segmentation of the target. To achieve this, we integrate a decoder based on the smooth network in [124], but in principle, any decoder used for image segmentation can be employed. The segmentation network in [124] has a Unet structure with a decoder block attached to each block in the ResNet backbone feature extractor as described in section 3.1. The fusion operation in the decoder block combines a residual layer with a channel attention layer [124].

In Paper E and Paper F, we modify the segmentation network by adding conditioning layers before the input to the decoder blocks. The conditioning is performed by concatenating the output of the target model with the feature blocks from the backbone network. These concatenated features are then processed using a small convnet for each feature block before they are passed to the decoder block. As a result, the features that are input to the decoder contain both generic feature information about the observed image and target-specific properties predicted by the target model. In the final step, the decoder outputs a single-channel prediction of the target segmentation. More details on the architecture can be found in Paper E and Paper F.

Bounding box supervision

In many practical applications, generating an accurate segmentation of the target is too costly. It is therefore desirable to also managing coarser annotations that are less demanding to produce. In Paper F, we propose an addition to the architecture to enable VOS using bound box supervision in the reference frame. To this end, we train a small network to encode a mask representation of the bounding box along with backbone features. The network outputs a target mask encoding, which lies in the same embedding space as the learned target labels that were discussed in section 3.3. Thus, we can exploit the already trained decoder network to predict the full target mask. VOS is then performed using the same procedure as when provided with mask annotations in the first frame. It is achieved by simply replacing the ground-truth mask y_0 with the predicted one. A favorable property of this approach is that the bounding box encoder network can be trained on image data in a separate phase, while the weights of the other network components can be kept frozen. This allows us to extend our VOS architecture to handle bounding box annotations in the reference frame without affecting the performance when segmentation masks are available.

Concluding remarks

This thesis presents works within time-of-flight phase unwrapping, point set registration, point set segmentation and video object segmentation.

In Paper A, we propose a method for multi-frequency phase unwrapping in amplitude modulated time-of-flight for depth imaging. The method estimates the likelihood of possible depths given a set of depth hypotheses. To this end, we compute a weighted KDE over all hypotheses in a spatial neighborhood of each pixel. The final depth is chosen as the one with the highest estimated density. Further, the density estimate can be used as a confidence measure to for instance suppress outliers. We find that the proposed phase unwrapping approach can improve the accuracy of depth cameras based on multi-frequency amplitude modulated time-of-flight and extends their effective range in both indoor and outdoor environments.

Paper B and Paper C discuss EM based point set registration. Specifically, Paper B addresses the problem of the inherent sensitivity to point sample density variations in probabilistic approaches. To address this problem, we propose to estimate the underlying distribution of the scene rather than directly model the distribution of the points. To this end, we derive a point-wise weighting based on the local density that allows us to minimize the Kullback-Leibner divergence between the density model and a latent scene distribution. Our method can therefore perform robust registration despite the presence of severe density variations, alleviating the need for re-sampling the point sets.

In Paper C, we investigate the incorporation of deep features into the EM based registration framework. To model the features, we propose to employ a von Mises-Fisher distribution, which we can efficiently optimize jointly with the other parameters of the mixture model. Importantly, our probabilistic formulation ensures that the full registration process is differentiable. This allows for learning features end-to-end, which significantly improves the performance of the approach. Moreover, by utilizing our end-to-end learning strategy we can learn a point-wise weighting instead of employing the pro-

posed density estimation approach in Paper B. As a result, our model learns to both handle density variations and highlight regions in the point sets that are useful for registration.

While we demonstrate improvements over the baselines in the experiments, there are still many challenges to be addressed in the proposed EM based frameworks. Currently, the methods struggle when the overlap between the point sets is low. The proposed point-wise weighting functions and learned features decrease the sensitivity, but low rates in overlap are still causing failures. We observe that the optimized objective occasionally pushes the registration methods to find the transformation parameters that maximize the overlap instead of finding the optimal alignment. By incorporating prior knowledge about the overlap, it should be possible to improve the performance of the proposed methods.

Further, in Paper D we propose a framework for 3D semantic segmentation. By projecting the 3D points into virtual camera views and render dense color and depth images, we can perform segmentation in the image space. The resulting pixel-wise segmentation scores are then re-projected into the point sets. Finally, the semantic label for each point is obtained by fusing scores from the different views. In our approach, we exploit the advantages of deep image segmentation conv-nets, which in comparison to 3D segmentation networks have much lower computational complexity and are generally applicable at a higher resolution. Furthermore, our approach allows us to perform pre-training on image datasets, thereby reducing the need for semantically labeled 3D point sets. An interesting direction to explore within this framework is to refine the selection of views such that the method becomes applicable to point sets of any structure.

Lastly, in Paper E and Paper F we investigate target appearance models within network architectures for semi-supervised video object segmentation. We propose a parametric target model, optimized during inference using the target mask obtained in the first frame. In the subsequent frames, we employ the learned target model to predict a coarse representation of the target in the image. Finally, the prediction is refined by a decoder network into a high-resolution segmentation mask. In paper Paper F, we further extend the idea by generalizing the representation of the target in the network. Instead of relying on the single-channel segmentation image as the target label, we propose to learn a label function that produces a multi-channel target representation. Importantly, by training the entire network end-to-end on annotated VOS data we can learn a label function, which generates a target representation that improves the segmentation accuracy.

There are many challenges in VOS that current state-of-the-art methods still cannot handle, such as high levels of object cluttering, long-term occlusions, and thin structures. Some of these challenges should be manageable if the VOS network also possesses geometric awareness. That includes the camera movement, the 3D shapes and transformations of the objects, and the 3D

---

structure of the scene. With such an awareness, the VOS network could learn useful priors about object movements and deformations and additionally be able to predict occlusions.

I believe the main limitation of the proposed learning-based methods in this thesis is their high dependency on annotated ground truth data. The manual labor required to generate datasets with ground truth labels limits the availability of large-scale datasets with high variability for many computer vision tasks. Specifically, for point set registration, point set segmentation and, video object segmentation, the shortage of large-scale and diverse datasets prevents learning-based methods from revealing their full capacity. Future works should therefore focus on exploring approaches that employ weaker supervision or even self-supervision during training.

Bibliography

- [1] Cyrus S. Bamji et al. “A 0.13 μm CMOS System-on-Chip for a 512×424 Time-of-Flight Image Sensor With Multi-Frequency Photo-Demodulation up to 130 MHz and 2 GS/s ADC”. In: *IEEE Journal on Solid-State Circuits* 50.1 (Jan. 2015) (pp. 20–22).
- [2] Linchao Bao, Baoyuan Wu, and Wei Liu. “CNN in MRF: Video Object Segmentation via Inference in A CNN-Based Higher-Order Spatio-Temporal MRF”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 5977–5986 (p. 39).
- [3] Maxim Berman, Amal Rannen Triki, and Matthew B Blaschko. “The Lovász-Softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 4413–4421 (p. 38).
- [4] Paul J. Besl and Neil D. McKay. “A Method for Registration of 3-D Shapes”. In: *IEEE TPAMI* 14.2 (1992), pp. 239–256 (p. 24).
- [5] Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. “Learning discriminative model prediction for tracking”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2019, pp. 6182–6191 (pp. 4, 5, 46).
- [6] Goutam Bhat, Felix Järemo Lawin, Martin Danelljan, Andreas Robinson, Michael Felsberg, Luc Van Gool, and Radu Timofte. “Learning what to learn for video object segmentation”. In: *16th European Conference on Computer Vision*. 2020 (pp. 4, 5, 171).
- [7] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006 (p. 25).

- [8] Alexandre Boulch, Joris Guerry, Bertrand Le Saux, and Nicolas Audebert. “SnapNet: 3D point cloud semantic labeling with 2D deep segmentation networks”. In: *Computers & Graphics* 71 (2018), pp. 189–198 (p. 42).
- [9] Sergi Caelles, K-K Maninis, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool. “One-shot video object segmentation”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE. 2017, pp. 5320–5329 (p. 45).
- [10] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. “Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs”. In: *IEEE transactions on pattern analysis and machine intelligence* 40.4 (2017), pp. 834–848 (p. 39).
- [11] Yang Chen and Gérard Medioni. “Object Modeling by Registration of Multiple Range Images”. In: *IEEE ICRA’91*. Apr. 1991 (p. 24).
- [12] Christopher Choy, Wei Dong, and Vladlen Koltun. “Deep Global Registration”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR’20)*. 2020 (pp. 5, 32).
- [13] Christopher Choy, JunYoung Gwak, and Silvio Savarese. “4D Spatio-Temporal ConvNets: Minkowski Convolutional Neural Networks”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019, pp. 3075–3084 (pp. 32, 40).
- [14] Christopher Choy, Jaesik Park, and Vladlen Koltun. “Fully Convolutional Geometric Features”. In: *ICCV*. 2019 (p. 32).
- [15] Haili Chui and Anand Rangarajan. “A feature registration framework using mixture models”. In: *Proceedings IEEE Workshop on Mathematical Methods in Biomedical Image Analysis. MMBIA-2000 (Cat. No. PR00737)*. IEEE. 2000, pp. 190–197 (p. 24).
- [16] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. “ATOM: Accurate Tracking by Overlap Maximization”. In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2019 (p. 46).
- [17] Martin Danelljan, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. “Eco: Efficient convolution operators for tracking”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 6638–6646 (p. 46).
- [18] Martin Danelljan, Giulia Meneghetti, Fahad Shahbaz Khan, and Michael Felsberg. “A Probabilistic Framework for Color-Based Point Set Registration”. In: *CVPR*. 2016 (pp. 6, 31, 33).

- [19] Martin Danelljan, Giulia Meneghetti, Fahad Shahbaz Khan, and Michael Felsberg. “Aligning the Dissimilar: A Probabilistic Method for Feature-Based Point Set Registration”. In: *ICPR*. 2016 (p. 31).
- [20] Haowen Deng, Tolga Birdal, and Slobodan Ilic. “3D local features for direct pairwise registration”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019, pp. 3244–3253 (p. 32).
- [21] Haowen Deng, Tolga Birdal, and Slobodan Ilic. “Ppf-foldnet: Unsupervised learning of rotation invariant 3d local descriptors”. In: *Proceedings of the European Conference on Computer Vision (ECCV)*. 2018, pp. 602–618 (p. 32).
- [22] Haowen Deng, Tolga Birdal, and Slobodan Ilic. “Ppfnet: Global context aware local features for robust 3d point matching”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 195–205 (p. 32).
- [23] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. “ImageNet: A Large-Scale Hierarchical Image Database”. In: *CVPR09*. 2009 (pp. 37, 38, 40, 44).
- [24] Nanqing Dong and Eric P Xing. “Few-Shot Semantic Segmentation with Prototype Learning.” In: *BMVC*. Vol. 3. 4. 2018 (p. 45).
- [25] David Droeschel, Dirk Holz, and Sven Behnke. “Multi-frequency Phase Unwrapping for Time-of-Flight Cameras”. In: *IEEE International Conference on Intelligent Robots and Systems*. 2010 (p. 20).
- [26] R. Durrett. *Probability: Theory and Examples*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2010. isbn: 9781139491136 (p. 30).
- [27] Yuval Eldar, Michael Lindenbaum, Moshe Porat, and Yehoshua Y Zeevi. “The farthest point strategy for progressive image sampling”. In: *TIP* 6.9 (1997), pp. 1305–1315 (pp. 28, 29).
- [28] Georgios D Evangelidis, Dionyssos Kounades-Bastian, Radu Horaud, and Emmanouil Z Psarakis. “A generative model for the joint registration of multiple point sets”. In: *European Conference on Computer Vision*. Springer. 2014, pp. 109–122 (pp. 6, 25–27, 30).
- [29] Georgios Evangelidis and Radu Horaud. “Joint Alignment of Multiple Point Sets with Batch and Incremental Expectation-Maximization”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40.6 (June 2018), pp. 1397–1410 (p. 27).
- [30] 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”. In: *International Journal of Computer Vision* 111.1 (Jan. 2015), pp. 98–136 (p. 37).

- [31] Chelsea Finn, Pieter Abbeel, and Sergey Levine. “Model-agnostic meta-learning for fast adaptation of deep networks”. In: *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org. 2017, pp. 1126–1135 (p. 45).
- [32] Martin A Fischler and Robert C Bolles. “Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography”. In: *Communications of the ACM* 24.6 (1981), pp. 381–395 (p. 31).
- [33] Mario Frank, Matthias Plaue, Holger Rapp, Ulrich Köthe, and Bernd Jähne. “Theoretical and experimental error analysis of continuous-wave time-of-flight range cameras”. In: *Optical Engineering* 48.1 (Jan. 2009) (pp. 6, 19).
- [34] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. “Dual attention network for scene segmentation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 3146–3154 (p. 39).
- [35] Kuniyiko Fukushima. “Neocognitron: A hierarchical neural network capable of visual pattern recognition”. In: *Neural networks* 1.2 (1988), pp. 119–130 (p. 3).
- [36] Andreas Geiger, Philip Lenz, and Raquel Urtasun. “Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite”. In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2012 (pp. 7, 32, 36).
- [37] Guy Godin, Marc Rioux, and Rejean Baribeau. “Three-dimensional registration using range and intensity information”. In: *Videometrics III*. Vol. 2350. International Society for Optics and Photonics. 1994, pp. 279–290 (pp. 30, 31).
- [38] Zan Gojcic, Caifa Zhou, Jan D. Wegner, Leonidas J. Guibas, and Tolga Birdal. “Learning multiview 3D point cloud registration”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR’20)*. 2020 (p. 32).
- [39] Steven Gold, Chien-Ping Lu, Anand Rangarajan, Suguna Pappu, and Eric Mjolsness. “New algorithms for 2D and 3D point matching: Pose estimation and correspondence”. In: *Advances in neural information processing systems* (1995), pp. 957–964 (p. 24).
- [40] Vladislav Golyanik, Sk Aziz Ali, and Didier Stricker. “Gravitational approach for point set registration”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 5802–5810 (p. 25).

- [41] Stephen Gould, Richard Hartley, and Dylan John Campbell. “Deep Declarative Networks”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2021) (p. 4).
- [42] Timo Hackel, Nikolay Savinov, Lubor Ladicky, Jan D Wegner, Konrad Schindler, and Marc Pollefeys. “SEMANTIC3D. NET: A NEW LARGE-SCALE POINT CLOUD CLASSIFICATION BENCHMARK.” In: *ISPRS Annals of Photogrammetry, Remote Sensing & Spatial Information Sciences* 4 (2017) (pp. 42–44).
- [43] Miles Hansard, Seungkyu Lee, Ouk Choi, and Radu Patrice Horaud. *Time-of-flight cameras: principles, methods and applications*. Springer Science & Business Media, 2012 (pp. 6, 19).
- [44] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. “Mask r-cnn”. In: *Proceedings of the IEEE international conference on computer vision*. 2017, pp. 2961–2969 (p. 44).
- [45] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778 (p. 4).
- [46] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification”. In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1026–1034 (p. 38).
- [47] Jeroen Hermans, Dirk Smeets, Dirk Vandermeulen, and Paul Suetens. “Robust point set registration using EM-ICP with information-theoretically optimal outlier handling”. In: *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE. 2011, pp. 2465–2472 (pp. 6, 24).
- [48] Magnus Rudolph Hestenes and Eduard Stiefel. *Methods of conjugate gradients for solving linear systems*. Vol. 49. 1. NBS Washington, DC, 1952 (p. 52).
- [49] Radu Horaud, Florence Forbes, Manuel Yguel, and Guillaume Dewaele and Jian Zhang. “Rigid and Articulated Point Registration with Expectation Conditional Maximization”. In: *PAMI* 33.3 (2011), pp. 587–602 (pp. 6, 24).
- [50] Berthold KP Horn, Hugh M Hilden, and Shahriar Negahdaripour. “Closed-form solution of absolute orientation using orthonormal matrices”. In: *JOSA A* 5.7 (1988), pp. 1127–1135 (p. 24).
- [51] Ping Hu, Gang Wang, Xiangfei Kong, Jason Kuen, and Yap-Peng Tan. “Motion-guided cascaded refinement network for video object segmentation”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 1400–1409 (p. 45).

- [52] Yuan-Ting Hu, Jia-Bin Huang, and Alexander G Schwing. “Video-match: Matching based video object segmentation”. In: *European Conference on Computer Vision*. Springer. 2018, pp. 56–73 (pp. 46, 47).
- [53] Bing Jian and Baba C. Vemuri. “Robust Point Set Registration Using Gaussian Mixture Models”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33.8 (2011), pp. 1633–1645 (p. 25).
- [54] Joakim Johnander, Martin Danelljan, Emil Brissman, Fahad Shahbaz Khan, and Michael Felsberg. “A Generative Appearance Model for End-to-end Video Object Segmentation”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019 (p. 46).
- [55] Joakim Johnander, Johan Edstedt, Martin Danelljan, Michael Felsberg, and Fahad Shahbaz Khan. “Deep Gaussian Processes for Few-Shot Segmentation”. In: *arXiv preprint arXiv:2103.16549* (2021) (p. 54).
- [56] Adrian P. P. Jongenelen, Donald G. Bailey, Andrew D. Payne, Adrian A. Dorrington, and Dale A. Carnegie. “Analysis of Errors in ToF Range Imaging with Dual-Frequency Modulation”. In: *IEEE transactions on instrumentation and measurement* 60.5 (May 2011) (p. 20).
- [57] Anna Khoreva, Rodrigo Benenson, Eddy Ilg, Thomas Brox, and Bernt Schiele. “Lucid Data Dreaming for Video Object Segmentation”. In: *International Journal of Computer Vision* 127.9 (Sept. 2019), pp. 1175–1197. issn: 1573-1405. doi: [10.1007/s11263-019-01164-6](https://doi.org/10.1007/s11263-019-01164-6). url: <https://doi.org/10.1007/s11263-019-01164-6> (p. 46).
- [58] Lena Klasén, Amanda Berg, Per-Erik Forssén, and Haibo Li. “Beyond 3D Imaging and Visualization”. In: *Proceedings of SSBA 2021 IAPR*. Non-reviewed workshop. SSBA. SSBA, Mar. 2021 (p. 18).
- [59] Abhijit Kundu, Xiaoqi Yin, Alireza Fathi, David Ross, Brian Brewington, Thomas Funkhouser, and Caroline Pantofaru. “Virtual multi-view fusion for 3d semantic segmentation”. In: *European Conference on Computer Vision*. Springer. 2020, pp. 518–535 (p. 42).
- [60] Felix Järemo Lawin, Martin Danelljan, Fahad Shahbaz Khan, Per-Erik Forssén, and Michael Felsberg. “Density adaptive point set registration”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 3829–3837 (p. 103).
- [61] Felix Järemo Lawin, Martin Danelljan, Patrik Tosteberg, Goutam Bhat, Fahad Shahbaz Khan, and Michael Felsberg. “Deep projective 3D semantic segmentation”. In: *International Conference on Computer Analysis of Images and Patterns*. Springer. 2017, pp. 95–107 (p. 139).

- [62] Felix Järemo Lawin and Per-Erik Forssén. “Registration Loss Learning for Deep Probabilistic Point Set Registration”. In: *International Virtual Conference on 3D Vision (3DV 2020)*. Nov. 2020 (pp. 5, 123).
- [63] Felix Järemo Lawin, Per-Erik Forssén, and Hannes Ovrén. “Efficient Multi-Frequency Phase Unwrapping using Kernel Density Estimation”. In: *European Conference on Computer Vision (ECCV)*. Amsterdam, Oct. 2016 (p. 77).
- [64] Yann LeCun, Yoshua Bengio, et al. “Convolutional networks for images, speech, and time series”. In: *The handbook of brain theory and neural networks* 3361.10 (1995), p. 1995 (p. 3).
- [65] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. “Deep learning”. In: *nature* 521.7553 (2015), pp. 436–444 (p. 3).
- [66] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. “Meta-Learning with Differentiable Convex Optimization”. In: *CVPR*. 2019 (p. 45).
- [67] Lei Li, Siyu Zhu, Hongbo Fu, Ping Tan, and Chiew-Lan Tai. “End-to-End Learning Local Multi-view Descriptors for 3D Point Clouds”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR’20)*. 2020 (p. 32).
- [68] Guosheng Lin, Anton Milan, Chunhua Shen, and Ian Reid. “Refinenet: Multi-path refinement networks for high-resolution semantic segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1925–1934 (p. 39).
- [69] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. “Microsoft coco: Common objects in context”. In: *European conference on computer vision*. Springer. 2014, pp. 740–755 (p. 37).
- [70] Ze Liu, Han Hu, Yue Cao, Zheng Zhang, and Xin Tong. “A closer look at local aggregation operators in point cloud analysis”. In: *European Conference on Computer Vision*. Springer. 2020, pp. 326–342 (p. 40).
- [71] Jonathan Long, Evan Shelhamer, and Trevor Darrell. “Fully convolutional networks for semantic segmentation”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 3431–3440 (pp. 5, 38, 39, 43).
- [72] Weixin Lu, Guowei Wan, Yao Zhou, Xiangyu Fu, Pengfei Yuan, and Shiyu Song. “DeepVCP: An End-to-End Deep Neural Network for Point Cloud Registration”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2019, pp. 12–21 (p. 32).

- [73] Martin Magnusson and et al. “Scan registration for autonomous mining vehicles using 3D-NDT”. In: *JFR* 24.10 (2007), pp. 803–827 (p. 25).
- [74] Kevis-Kokitsi Maninis, Sergi Caelles, Yuhua Chen, Jordi Pont-Tuset, Laura Leal-Taixé, Daniel Cremers, and Luc Van Gool. “Video object segmentation without temporal information”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* (2018) (p. 45).
- [75] Kanti V. Mardia and Peter E. Jupp. “Directional Statistics”. In: John Wiley and Sons Inc., 2000. Chap. 10.3.1 (p. 34).
- [76] Donald W Marquardt. “An algorithm for least-squares estimation of nonlinear parameters”. In: *Journal of the society for Industrial and Applied Mathematics* 11.2 (1963), pp. 431–441 (p. 52).
- [77] Martin Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. url: <https://www.tensorflow.org/> (p. 37).
- [78] Daniel Maturana and Sebastian Scherer. “Voxnet: A 3d convolutional neural network for real-time object recognition”. In: *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2015, pp. 922–928 (pp. 32, 40).
- [79] Hao Men, Biruk Gebre, and Kishore Pochiraju. “Color point cloud registration with 4D ICP algorithm”. In: *2011 IEEE International Conference on Robotics and Automation*. IEEE. 2011, pp. 1511–1516 (pp. 30, 31).
- [80] Microsoft Corporation. *Microsoft Kinect for Windows SDK (SDK for Kinect v2)*. <http://www.microsoft.com/en-us/download/details.aspx?id=44561>. 2015 (p. 20).
- [81] Kevin P. Murphy. *Machine Learning: A Probabilistic Perspective*. MIT Press, 2012 (p. 25).
- [82] Andriy Myronenko and Xubo Song. “Point set registration: Coherent point drift”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.12 (2010), pp. 2262–2275 (pp. 6, 24–27).

- [83] Klas Nordberg, Gösta Granlund, and Hans Knutsson. “Representation and learning of invariance”. In: *Proceedings of 1st International Conference on Image Processing*. Vol. 2. IEEE. 1994, pp. 585–589 (p. 3).
- [84] Jens Ogniewski and Per-Erik Forssén. “Pushing the Limits for View Prediction in Video Coding”. In: *12th International Conference on Computer Vision Theory and Applications (VISAPP’17)*. Porto, Portugal: Scitepress Digital Library, Feb. 2017 (p. 42).
- [85] Seoung Wug Oh, Joon-Young Lee, Kalyan Sunkavalli, and Seon Joo Kim. “Fast video object segmentation by reference-guided mask propagation”. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE. 2018, pp. 7376–7385 (p. 46).
- [86] Seoung Wug Oh, Joon-Young Lee, Ning Xu, and Seon Joo Kim. “Video object segmentation using space-time memory networks”. In: *Proceedings of the IEEE International Conference on Computer Vision (2019)* (pp. 46, 47).
- [87] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. “PyTorch: An Imperative Style, High-Performance Deep Learning Library”. In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, and R. Garnett. Curran Associates, Inc., 2019, pp. 8024–8035. url: <http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf> (p. 37).
- [88] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung. “A Benchmark Dataset and Evaluation Methodology for Video Object Segmentation”. In: *Computer Vision and Pattern Recognition*. 2016 (pp. 4, 44, 50).
- [89] Federico Perazzi, Anna Khoreva, Rodrigo Benenson, Bernt Schiele, and Alexander Sorkine-Hornung. “Learning video object segmentation from static images”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 2663–2672 (p. 45).
- [90] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas 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*. 2017, pp. 652–660 (pp. 32, 40).

- [91] Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. “PointNet++ deep hierarchical feature learning on point sets in a metric space”. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 2017, pp. 5105–5114 (p. 40).
- [92] Herbert Robbins and Sutton Monro. “A stochastic approximation method”. In: *The annals of mathematical statistics* (1951), pp. 400–407 (p. 4).
- [93] Andreas Robinson, Felix Järemo Lawin, Martin Danelljan, Fahad Shahbaz Khan, and Michael Felsberg. “Learning fast and robust target models for video object segmentation”. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2020, pp. 7406–7415 (p. 155).
- [94] Murray Rosenblatt. “Remarks on Some Nonparametric Estimates of a Density Function”. In: *Annals of Mathematical Statistics* 27.3 (1956), pp. 832–837 (p. 3).
- [95] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. “Learning representations by back-propagating errors”. In: *nature* 323.6088 (1986), pp. 533–536 (p. 4).
- [96] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. “Fast Point Feature Histograms (FPFH) for 3D Registration”. In: *IEEE International Conference on Robotics and Automation (ICRA’09)*. Kobe, Japan, May 2009 (pp. 31, 32).
- [97] Radu Bogdan Rusu, Zoltan Csaba Marton, Nico Blodow, and Michael Beetz. “Learning informative point classes for the acquisition of object model maps”. In: *2008 10th International Conference on Control, Automation, Robotics and Vision*. IEEE. 2008, pp. 643–650 (p. 32).
- [98] Joaquim Salvi, Jordi Pages, and Joan Batlle. “Pattern codification strategies in structured light systems”. In: *Pattern recognition* 37.4 (2004), pp. 827–849 (p. 18).
- [99] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. “The graph neural network model”. In: *IEEE transactions on neural networks* 20.1 (2008), pp. 61–80 (p. 32).
- [100] Florian Schroff, Dmitry Kalenichenko, and James Philbin. “FaceNet: A Unified Embedding for Face Recognition and Clustering”. In: *CVPR’15*. 2015 (p. 32).
- [101] Aleksandr Segal, Dirk Hähnel, and Sebastian Thrun. “Generalized-ICP”. In: *RSS*. 2009 (p. 24).
- [102] Karen Simonyan and Andrew Zisserman. “Very Deep Convolutional Networks for Large-Scale Image Recognition”. In: *International Conference on Learning Representations*. 2015 (pp. 38, 43).

- [103] Richard Sinkhorn and Paul Knopp. “Concerning nonnegative matrices and doubly stochastic matrices”. In: *Pacific Journal of Mathematics* 21.2 (1967), pp. 343–348 (p. 24).
- [104] Shuochen Su, Felix Heide, Gordon Wetzstein, and Wolfgang Heidrich. “Deep end-to-end time-of-flight imaging”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 6383–6392 (p. 20).
- [105] Richard Szeliski. *Computer Vision: Algorithms and Applications*. Springer-Verlag New York, Inc., 2010 (p. 42).
- [106] Anderson CM Tavares, Felix Järemo Lawin, and Per-Erik Forssén. “Assessing Losses for Point Set Registration”. In: *IEEE Robotics and Automation Letters* 5.2 (2020), pp. 3360–3367 (p. 25).
- [107] Prune Truong, Martin Danelljan, Luc V Gool, and Radu Timofte. “GO-Cor: Bringing Globally Optimized Correspondence Volumes into Your Neural Network”. In: *Advances in Neural Information Processing Systems* 33 (2020) (p. 54).
- [108] Yanghai Tsin and Takeo Kanade. “A correlation-based approach to robust point set registration”. In: *European conference on computer vision*. Springer. 2004, pp. 558–569 (p. 25).
- [109] Jonas Unger, Andrew Gardner, Per Larsson, and Francesco Banterle. “Capturing Reality for Computer Graphics Applications”. In: *Sigraph Asia Course*. 2015 (pp. 29, 31).
- [110] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. “Attention is all you need”. In: *arXiv preprint arXiv:1706.03762* (2017) (pp. 4, 46).
- [111] A. Vedaldi and K. Lenc. “MatConvNet – Convolutional Neural Networks for MATLAB”. In: *Proceeding of the ACM Int. Conf. on Multimedia*. 2015 (p. 37).
- [112] Paul Voigtlaender and Bastian Leibe. “FEELVOS: Fast End-to-End Embedding Learning for Video Object Segmentation”. In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019 (pp. 46, 47).
- [113] Michael W Walker, Lejun Shao, and Richard A Volz. “Estimating 3-D location parameters using dual number quaternions”. In: *CVGIP: image understanding* 54.3 (1991), pp. 358–367 (p. 24).
- [114] Yue Wang and Justin M Solomon. “Deep closest point: Learning representations for point cloud registration”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2019, pp. 3523–3532 (p. 32).

- [115] Yue Wang and Justin M Solomon. “PRNet: Self-Supervised Learning for Partial-to-Partial Registration”. In: *Advances in Neural Information Processing Systems*. 2019, pp. 8812–8824 (p. 32).
- [116] Zhaoyang Wang, Hua Du, Seungbae Park, and Huimin Xie. “Three-dimensional shape measurement with a fast and accurate approach”. In: *Applied optics* 48.6 (2009), pp. 1052–1061 (p. 20).
- [117] Ziqin Wang, Jun Xu, Li Liu, Fan Zhu, and Ling Shao. “Ranet: Ranking attention network for fast video object segmentation”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2019, pp. 3978–3987 (p. 46).
- [118] Lingzhu Xiang, Florian Echtler, Christian Kerl, Thiemo Wiedemeyer, Lars, hanyazou, Ryan Gordon, Francisco Facioni, laborer2008, Rich Wareham, Matthias Goldhoorn, alberth, gaborpapp, Steffen Fuchs, jmtatsch, Joshua Blake, Federico, Henning Jungkurth, Yuan Mingze, vinouz, Dave Coleman, Brendan Burns, Rahul Rawat, Serguei Mokhov, Paul Reynolds, P.E. Viau, Matthieu Fraissinet-Tachet, Ludique, James Billingham, and Alistair. *libfreenect2: Release 0.2*. Apr. 2016. doi: [10.5281/zenodo.50641](https://doi.org/10.5281/zenodo.50641). url: <https://doi.org/10.5281/zenodo.50641> (p. 20).
- [119] Ning Xu, Linjie Yang, Yuchen Fan, Jianchao Yang, Dingcheng Yue, Yuchen Liang, Brian Price, Scott Cohen, and Thomas Huang. “YouTube-VOS: Sequence-to-Sequence Video Object Segmentation”. In: *European Conference on Computer Vision*. Springer. 2018, pp. 603–619 (pp. 44, 45).
- [120] Linjie Yang, Yanran Wang, Xuehan Xiong, Jianchao Yang, and Aggelos K Katsaggelos. “Efficient video object segmentation via network modulation”. In: *algorithms* 29 (2018), p. 15 (p. 46).
- [121] Zongxin Yang, Yunchao Wei, and Yi Yang. “Collaborative video object segmentation by foreground-background integration”. In: *Proceedings of the European Conference on Computer Vision*. 2020 (pp. 46, 47).
- [122] Zi Jian Yew and Gim Hee Lee. “3DFeat-Net: Weakly Supervised Local 3D Features for Point Cloud Registration”. In: *ECCV*. 2018 (p. 32).
- [123] Zi Jian Yew and Gim Hee Lee. “Rpm-net: Robust point matching using learned features”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020, pp. 11824–11833 (pp. 31, 32).
- [124] Changqian Yu, Jingbo Wang, Chao Peng, Changxin Gao, Gang Yu, and Nong Sang. “Learning a discriminative feature network for semantic segmentation”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 1857–1866 (pp. 39, 55).

- [125] Wentao Yuan, Benjamin Eckart, Kihwan Kim, Varun Jampani, Dieter Fox, and Jan Kautz. “DeepGMR: Learning Latent Gaussian Mixture Models for Registration”. In: *European Conference on Computer Vision*. Springer. 2020, pp. 733–750 (p. 31).
- [126] Andy Zeng, Shuran Song, Matthias Nießner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser. “3DMatch: Learning Local Geometric Descriptors from RGB-D Reconstructions”. In: *CVPR*. 2017 (pp. 18, 32, 36).
- [127] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. “Pyramid scene parsing network”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 2881–2890 (p. 39).
- [128] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. “Scene Parsing through ADE20K Dataset”. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017 (pp. 37, 40, 44).

Part II

Publications

Papers

The papers associated with this thesis have been removed for copyright reasons. For more details about these see:

<http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-176054>

FACULTY OF SCIENCE AND ENGINEERING

Linköping Studies in Science and Technology. Dissertations, No. 2151, 2021
Department of Electrical Engineering

Linköping University
SE-581 83 Linköping, Sweden

www.liu.se