Effective Representations for Road Scene Understanding with Scalable Learning

Tom Adriaan Hubert Bruls

Keble College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Trinity 2020

Effective Representations for Road Scene Understanding with Scalable Learning

Candidate: Tom Adriaan Hubert Bruls MSc

Supervisor: Professor Paul Newman

Examiners: Professor Nick Hawes & Professor Eduardo Nebot

Date of Examination: 4 November 2020

University of Oxford

Mobile Robotics Group

Oxford Robotics Institute

Department of Engineering Science

Acknowledgements

Firstly, I would like to express my deepest gratitude to my supervisor Professor Paul Newman. His optimism, encouragement, and insights have not only shaped me as a researcher but also as a person. I will forever be grateful that he believed in me as a promising researcher and gifted me the opportunity to develop myself at his group and this institution. I would like to thank him sincerely for his care on a personal level and including me into the ORI family.

I would like to pay my special regards to Dr Will Maddern and Dr Akshay Morye whose guidance has been invaluable during the beginning of my DPhil. They quickly taught me how to be successful on a day-to-day basis in the wonderful but challenging world of academic research and stimulated me to explore outside of that world as well. On a similar note, I would like to thank Dr Lars Kunze for our academic discussions and for his guidance in formulating my research, my presentations, and this thesis.

I would like to thank my co-authors Dr Paul Amayo, Dr Tarlan Suleymanov, and especially Horia Porav. It has been extremely inspiring to collaborate with him, a true "research engineer" in the purest way. Our projects together have taught me the enormous benefits of possessing useful skills in the entire engineering spectrum from low-level hardware to high-level research. I express my specific gratitude to Liesbeth Bruls, Luuk Bruls, and Valentina-Nicoleta Musat for their hours of work helping me with the extremely tedious process of pixel-wise labelling of road markings for evaluation purposes. Furthermore, I would like to thank everyone else at ORI who has made this thesis and the journey of my DPhil possible and enjoyable by either supporting my work or me personally.

I would like to thank my peer students in Room 3 in particular Kevin, Marlin, Rowan, Sarah, and Simon for their academic insights and feedback, but more importantly for our social chats. They have made sure that my DPhil has been more than just an academic experience by connecting with me on a personal level as well.

Tenslotte, wil ik mijn ouders Liesbeth en André, en mijn broer Luuk bedanken. Zij hebben met hun steun en begrip er altijd voor gezorgd dat ik mij volledig kon toeleggen op mijn promotieonderzoek en mijn persoonlijke ontwikkeling.

Abstract

Autonomous vehicles require an accurate understanding of the scene for safe operation in real-world driving scenarios. This thesis examines and offers effective representations for road scene understanding based on in-situ perception, which can be employed directly for planning and decision making in a variety of complex urban environments and under wide-ranging environmental conditions.

A common, versatile scene representation is the pixel-wise semantic segmentation obtained by deep neural networks. However, this representation is limited in its direct usefulness for several reasons. Firstly, the pixel-wise semantic segmentation does not naturally support the high-level reasoning required for complex driving manoeuvres. This thesis resolves that limitation by focussing on a hierarchical, graph-based representation, the *scene graph*, which combines segmented entities and object-centric perception in bird's-eye view at a suitable abstraction level for decision making. The road markings are a crucial prerequisite in this representation as their underlying meaning dictates the desired driving behaviour. Secondly, semantic segmentation often lacks this semantic understanding of the road markings due to the inordinate cost of labelling adequate training data. Instead, this thesis presents and compares a model-driven and data-driven approach for self-supervised road marking classification. Whereas the former leverages additional sensor modalities and domain knowledge, the latter employs state-of-the-art image-to-image translation techniques to synthesize training data. Thirdly, semantic segmentation is commonly performed in the front-facing perspective, which does not explicitly encode distances or directly link to the vehicle's action space. We tackle this problem by learning an improved bird's-eye-view mapping, called "boosted IPM", which aids scene graph generation in real-world scenarios.

In addressing the above, we introduce scalable, self-supervised learning techniques by employing design principles such as transfer learning, leveraging domain knowledge, and data synthesis. Furthermore, we improve the robustness of the representations under wide-ranging environmental conditions by image restoration and learning appearance-invariant representations. The presented methodologies serve as a valuable starting point to devise effective and efficient representations for road scene understanding based on in-situ perception in real-world scenarios.

Contents

List of Figures **xi**

List of Tables **xiii**

List of Abbreviations **xv**

1 Introduction **1**
- 1.1 Motivation . . . 1
- 1.2 Themes . . . 4
- 1.3 Contributions . . . 7
- 1.4 Thesis Outline . . . 8
- 1.5 Publications . . . 10

2 Representations & Data for Road Scene Understanding **13**
- 2.1 Representations for Road Scene Understanding . . . 13
  - 2.1.1 Effective Representations . . . 14
  - 2.1.2 Non-Parametric Representations . . . 15
  - 2.1.3 Hybrid Representations . . . 17
- 2.2 Data for Road Scene Understanding . . . 19
  - 2.2.1 Manually-Labelled Datasets . . . 20
  - 2.2.2 Physics-Based Simulators . . . 20
  - 2.2.3 Unlabelled Datasets . . . 21

3 Self-Supervised Scene Segmentation under Wide-Ranging Environmental Conditions **25**
- 3.1 Publication . . . 27
- 3.2 Summary of the Results . . . 36
- 3.3 Conclusion . . . 37

4 Representations for Integration: Scene Graphs **39**
- 4.1 Hierarchical Road Scene Understanding . . . 40
- 4.2 Prerequisites for Scene Graphs . . . 42
  - 4.2.1 Representations for Conduct . . . 42
  - 4.2.2 Representations for Overview . . . 45

5 Representations for Conduct I: Road Marking Segmentation 49
5.1 Publication . . . 54
5.2 Approximated Road Marking Labels . . . 63
5.2.1 Further Details . . . 63
5.2.2 Further Results . . . 65
5.2.3 Further Discussion . . . 66
5.3 DNN Road Marking Segmentation . . . 67
5.3.1 Further Details . . . 67
5.3.2 Further Results . . . 68
5.3.3 Further Discussion . . . 72
5.4 DNN Road Marking Segmentation under Rainy Conditions . . . 73
5.4.1 De-Raining Images . . . 73
5.4.2 Summary of the Results . . . 74
5.5 Conclusion . . . 76

6 Representations for Conduct II: Road Marking Classification 79
6.1 Model-Driven Road Marking Classification . . . 82
6.2 Data-Driven Road Marking Classification . . . 83
6.2.1 Publication . . . 84
6.2.2 Further Details . . . 93
6.2.3 Further Results . . . 95
6.2.4 Further Discussion . . . 96
6.3 Qualitative Comparison . . . 97
6.4 Conclusion . . . 101

7 Representations for Overview: Boosted Inverse Perspective Mapping 105
7.1 Publication . . . 108
7.2 Further Details . . . 117
7.3 Further Results . . . 117
7.3.1 Boosted IPM . . . 118
7.3.2 Road Marking Segmentation in Bird's-Eye View . . . 124
7.4 Further Discussion . . . 129
7.5 Conclusion . . . 130

8 Summary and Future Directions 133
8.1 Summary . . . 133
8.2 Broader Impact . . . 136
8.3 Future Directions . . . 137
8.3.1 Effective Representations for Road Scene Understanding . . . 137

8.3.2 Scalable Learning for Road Scene Understanding 138
8.3.3 Appearance Invariance for Road Scene Understanding 140
8.4 Closing Remarks 141

References **143**

Appendices

A Reading between the Lanes: Road Layout Reconstruction from Partially Segmented Scenes **155**

B I Can See Clearly Now: Image Restoration via De-Raining **165**

C Semantic Classification of Road Markings from Geometric Primitives **175**

List of Figures

1.1 The mediated approach for autonomous driving in urban environments. 2
1.2 Semantic segmentation of an urban road scene. 3

2.1 A non-parametric representation for describing the road topology in the presence of occlusions. 16
2.2 A parametric representation for describing complex urban road layouts. 18
2.3 A graph-based representation for describing complex urban road layouts. 19
2.4 Manually-labelled road marking datasets. 21
2.5 A bird's-eye-view dataset captured in the CARLA simulator. 21
2.6 The Oxford RobotCar platform and dataset. 22
2.7 Automatically-generated training pairs by employing data synthesis. 23

4.1 An example of a generated scene graph. 41
4.2 Road marking representations. 43
4.3 Semantic road marking classification. 44
4.4 IPM for obtaining a bird's-eye view. 45
4.5 Challenging scenarios for generating scene graphs. 46

5.1 Road marking segmentation and an approximated training label for an urban road scene. 50
5.2 Challenging scenarios for road marking segmentation. 51
5.3 Road surface extraction by leveraging LiDARs. 63
5.4 Enhanced road surface extraction to improve the approximated road marking labels. 64
5.5 Correct generation of approximated road marking labels by leveraging LiDAR reflectance. 65
5.6 Approximated road marking labels under challenging conditions . . 65
5.7 De-raining images to improve road marking segmentation. 74

6.1 Leveraging contextual information to aid road marking classification. 80
6.2 Model-driven road marking classification under various conditions. . 83
6.3 Road marking classification under favourable conditions. 98

6.4 The difference in the output representation between the model-driven and data-driven approach leading to spatial inconsistencies. 98
6.5 Temporal inconsistencies arising when road markings are classified on a per-frame basis. 99
6.6 The difference between the model-driven and data-driven approach with regard to the classification of symbols and letters. 100

7.1 An automatically-generated training label for boosted IPM. 117
7.2 Additional variants of the boosted IPM framework. 118
7.3 A comparison of full image PSNR as a function of the distance for the various types of IPM. 123
7.4 A comparison of road surface PSNR as a function of the distance for the various types of IPM. 123
7.5 The boosted Road Layout framework. 125

8.1 Novel weather and lighting combinations synthesized from available data. 140

List of Tables

5.1 The regularization effect of dropout for road marking segmentation. 67
5.2 Road marking segmentation in CamVid and Rainy RobotCar by pretraining on approximated road marking labels. 70
5.3 Road marking segmentation in CamVid by pretraining on approximated road marking versus Cityscapes labels. 70
5.4 Road marking segmentation in Rainy RobotCar by adding approximated road marking labels. 72

6.1 A qualitative comparison of frequency and total balancing for the classification of zig-zag road markings. 93
6.2 A qualitative comparison of the classification of bus stop road markings. 94
6.3 A qualitative comparison of the classification of diagonal road markings. 94
6.4 A qualitative comparison of the classification of warning triangles. . 95
6.5 A qualitative comparison of the classification of zig-zag road markings. 96
6.6 A comparison of different image-to-image translation frameworks for synthesizing bus stop road markings. 96
6.7 A comparison of different image-to-image translation frameworks for synthesizing diagonal road markings. 97

7.1 A qualitative comparison of various types of IPM in the CARLA simulator. 120
7.2 A quantitative comparison of various types of IPM in the CARLA simulator. 122
7.3 A qualitative comparison of road marking segmentation in the various types of IPM in the CARLA simulator. 126
7.4 A quantitative comparison of road marking segmentation in the various types of IPM in the CARLA simulator. 128

List of Abbreviations

2D/3D . . . . Two-/Three-Dimensional

Acc . . . . . Accuracy

CNN . . . . . Convolutional Neural Network

CORAL . . . . COnvex Relaxation ALgorithm

CRF . . . . . Conditional Random Field

DNN . . . . . Deep Neural Network

CV . . . . . CamVid (dataset)

FM . . . . . Feature Matching

GAN . . . . . Generative Adversarial Network

GPU . . . . . Graphics Processing Unit

LSTM . . . . . Long Short-Term Memory

(m)IoU . . . . (mean) Intersection over Union

IPM . . . . . Inverse Perspective Mapping

LiDAR . . . . Light Detection And Ranging

MRG . . . . . Mobile Robotics Group

ORI . . . . . Oxford Robotics Institute

Pre . . . . . Precision

PSNR . . . . . Peak Signal-to-Noise Ratio

RANSAC . . . RANdom SAmpling Consensus

Rec . . . . . Recall

RGB . . . . . Red Green Blue

RL . . . . . (boosted) Road Layout

RC . . . . . (Oxford) RobotCar (dataset)

RR . . . . . (Oxford) Rainy RobotCar (dataset)

SSIM . . . . . Structural SIMilarity

UK . . . . . United Kingdom

VGG . . . . . Visual Geometry Group

VO . . . . . Visual Odometry

1 Introduction

Contents

1.1 Motivation **1**
1.2 Themes **4**
1.3 Contributions **7**
1.4 Thesis Outline **8**
1.5 Publications **10**

1.1 Motivation

Autonomous vehicles are transforming transportation as we know it today. One of the most appealing prospects of an autonomous future is safer roads. It is estimated that 94% of fatal accidents are due to human error [1]. Therefore, eliminating the driver has become a top priority in automotive design. Although some manufacturers such as Waymo are running geofenced tests with actual passengers, deploying autonomous vehicles everywhere and at any time remains a formidable and open challenge. Most of the implementations available to a larger audience are solely designed for high-way environments such as the Tesla Autopilot [2]. In contrast, urban autonomous driving has experienced slower progress. Reasons for this are the significantly more complex road layouts, which are necessary in order to serve

(a) **(b)**

Figure 1.1: Autonomous vehicles have to manoeuvre through complex road layouts shared by different types of traffic participants in urban environments, **(a)**. The mediated approach builds a representation which describes all relevant (road) objects in its environment and can be employed for decision making, **(b)**. Photographs **(a)** and **(b)** courtesy of Waymo.

different types of traffic participants, and the fact that the necessary techniques still form active research topics in their respective fields.

Two competing paradigms currently exist to accomplish urban autonomous driving: the behavioural and the mediated approach. Although the behavioural approach [3]–[5] (i.e. end-to-end/deep driving) is actively researched, most (if not all) real-world implementations follow the mediated approach. The latter consists of a more traditional robotics pipeline that involves the following steps performed in a continuous loop: perception, state estimation (i.e. localization and scene understanding), planning, and decision making (i.e. action selection). Firstly, the vehicle uses its sensor suite to perceive the environment. Secondly, it builds a representation of the scene, including all relevant road objects, and localizes itself within this representation. Finally, planning and decision making based on this representation and the ego location allows the vehicle to navigate safely and efficiently to its desired destination.

This thesis focusses on understanding urban road scenes within such a pipeline, as shown in Figure 1.1. Road scene understanding is the task of inferring complex scene and road layouts from perception inputs such that the resulting representation can be employed for planning and decision making. While the availability of HD-maps, which provide the rich information necessary for autonomous driving, has increased drastically with the rise of offline map providers (e.g. HERE and TomTom), these

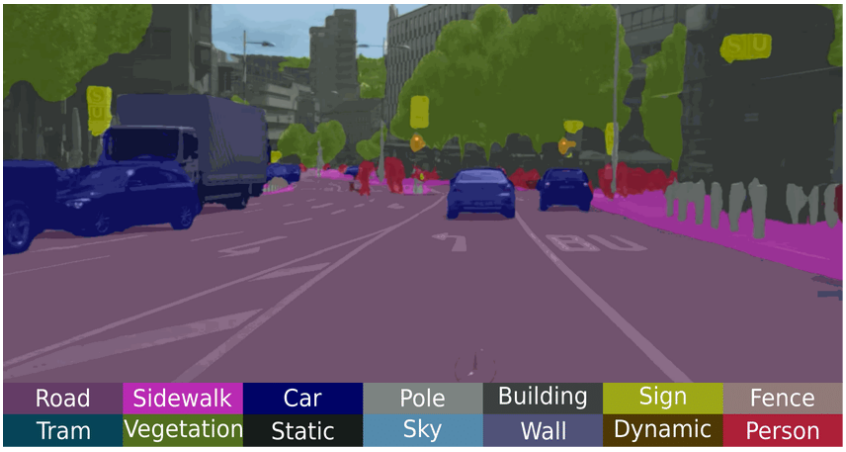


Figure 1.2: Semantic segmentation of an urban road scene [9]. Every pixel is classified and coloured according to its semantic class. High performance can be achieved under favourable conditions due to substantial progress in deep learning. However, the pixel-wise output representation is limited in its direct usefulness during real-world deployment for multiple reasons, some of which are addressed in this thesis.

do not entirely negate the need for in-situ perception. Urban infrastructure changes continually, which might lead to substantial differences between the real world and the map, thereby increasing the cost of maintaining accurate maps [6]. Although we acknowledge the benefits of leveraging available maps, we choose to focus on in-situ perception to bridge the gap towards real-world deployment everywhere and at any time. Extensive sensor suites, including but not limited to different types of imaging cameras, LiDARs, and radars, are commonly used for this task [7], [8]. While the presented frameworks in this thesis are bootstrapped by leveraging different sensor modalities, they only require a monocular camera during deployment, ensuring that the presented work remains versatile and cost-efficient on a large scale.

Image perception has experienced tremendous progress in the last couple of years with the emergence of deep learning techniques in computer vision. By using an abundance of data, more powerful computing resources, and novel learning frameworks, numerous perception tasks such as object detection, depth estimation, and semantic segmentation have progressed towards real-world deployment. Urban road layouts are typically complex, vary substantially, and are often partly occluded. Consequently, non-parametric models such as deep semantic segmentation networks have attracted a great deal of interest, which has led to impressive results for

pixel-wise scene understanding of images [10], [11], as exemplified in Figure 1.2. The output is visually appealing and gives the user the idea that the scene is fully understood and thus the problem is solved. However, in reality, its direct usefulness for decision making and planning in autonomous driving is limited for several reasons.

Firstly, the segmentation generally does not include a semantic understanding of the road markings in the scene. This information is crucial for decision making as the road markings impose the road rules and guide the traffic participants. Incorporating such small object classes in the datasets is difficult and costly, though, because the pixel-wise labels are manually annotated in general. Although some datasets [12], [13] include a single road marking class, they fail to provide the semantic meaning of the various types. Only the recently-published ApolloScape dataset [14] and VPGNet [15] include this level of detail. However, the labels require manual annotation which does not scale efficiently to new environments. Secondly, the pixel-wise output representation does not directly allow for high-level reasoning concerning driving actions. Although it is possible to learn driving policies directly from the semantic segmentation [16], this does not provide the interpretability in the decision-making process necessary for safe and reliable real-world deployment. Lastly, semantic scene segmentation is commonly performed in the front-facing camera. Nevertheless, it is difficult to relate this view to the vehicle's action space due to its perspective; therefore, many autonomous driving tasks are commonly performed in bird's-eye view.

This thesis aims to resolve these limitations by devising more effective representations for road scene understanding based on in-situ perception, which can be employed directly for planning and decision making in various complex urban environments and under wide-ranging environmental conditions.

1.2 Themes

The work in this thesis follows three underlying themes at a high level in accordance with the aforementioned goal.

Theme 1: Effective Representations for Road Scene Understanding

We aim to design representations that can be linked directly to the action space of the autonomous vehicle. As discussed previously, the pixel-wise representation obtained by deep semantic segmentation networks does not suffice for this purpose. We resolve the aforementioned limitations in the following ways:

(a) **Representations for Integration**. We demonstrate hybrid frameworks that combine object-centric perception with (pixel-wise) learning techniques and are informed by domain knowledge. The resulting representations offer a suitable abstraction level to aid decision making while maintaining the versatility to describe a wide range of traffic situations.

(b) **Representations for Conduct**. We extend the level of detail of the general semantic segmentation representation significantly by including a semantic understanding of the road markings. This extension captures the road rules of the scene, which dictate the rules of conduct for the traffic participants.

(c) **Representations for Overview**. We adjust the perspective of the images and their semantic interpretations from the front-facing camera towards a bird's-eye view. This provides a more intuitive overview of the traffic situation, which is more convenient for reasoning as it explicitly encodes distances, making it more closely linked to the vehicle's action space.

Theme 2: Scalable Learning for Road Scene Understanding

In order to scale and quickly adapt to the wide range of urban environments that are encountered during real-world deployment, it is necessary to implement data-driven approaches instead of purely model-driven ones, which offer limited versatility. Nevertheless, data-driven approaches require vast quantities of expensive, labelled data to generalize across all conditions and environments encountered during real-world deployment. At this scale, the common practice of manually annotating these data samples is infeasible.

Therefore, we aim to reduce the labelling effort for road scene understanding tasks throughout this thesis by employing *self-supervised learning*. In order to generate image training pairs for urban scenes automatically, we employ the following three techniques:

(a) **Transfer Learning**. We transfer knowledge from related domains, complementary sensor modalities, or additional expert systems towards the domain and modality of interest.

(b) **Leveraging Domain Knowledge**. We leverage domain knowledge, which is available in abundance since roads are constructed according to well-defined definitions.

(c) **Data Augmentation and Synthesis**. We employ state-of-the-art image-to-image translation techniques to augment and synthesize new training pairs.

Theme 3: Appearance Invariance for Road Scene Understanding

For autonomous vehicles to operate safely in the real world, their deployed algorithms must be robust to changes in environmental (i.e weather and lighting) conditions. Despite this, progress in semantic segmentation is generally evaluated under favourable conditions, which is neglectful of the challenges faced during deployment. Adverse weather conditions often lead to occlusions (e.g. raindrops on the lens or overexposure), limiting the reasoning in particular image regions. We aim to restore these regions to allow for robust scene understanding and employ the following techniques to accomplish that:

(a) **Learning Appearance-Invariant Representations**. We steer networks to learn appearance-invariant representations that "strip" the appearance from the image and are optimized for the task of interest. By splitting these two tasks explicitly, we learn more general semantic representations.

(b) **Image Restoration**. We employ DNNs to remove occlusions by training against the corresponding clear image.

In this third theme, the author worked in close collaboration with Horia Porav.

1.3 Contributions

Considering the themes discussed above, this thesis makes the following principal contributions:

- The concept of underlying appearance-invariant representations, which comprise all necessary information for decision making and other relevant tasks in autonomous driving (T-3a) [17].

- A self-supervised approach for binary road marking segmentation under various conditions as a step towards semantic classification [18] (T-1b). Complementary sensor modalities and domain knowledge are leveraged to generate training data automatically (T-2a and T-2b).

- An integration and discussion of the obtained binary road markings into a higher-level scene understanding framework for inferring the road layout (T-1b) [19].

- A data-driven approach for road marking classification using the obtained binary segmentation to synthesize photo-realistic training images for predefined labels (T-2b and T-2c) [20]. Vast quantities of images containing rare road marking classes are synthesized and used for training combined with a newly-introduced class-weighted loss function to boost performance (T-1b).

- A framework for learning an improved IPM, which is beneficial for high-level scene understanding (T-1a, T-1b, and T-1c) [21]. The training data is generated automatically from the sensor calibrations and VO (T-2a). This naturally leads to improvements in the presence of both occlusions and extreme illumination (T-3b).

Additionally, this thesis makes the following supporting contributions in collaboration:

- A demonstration of self-supervised road scene semantic segmentation under wide-ranging environmental conditions [17]. An image-to-image translation technique is employed to change the appearance of the images and thus generate training data automatically (T-2c). A framework of input adapters is designed to explicitly learn an appearance-invariant representation optimal for semantic segmentation (T-3a).

- A demonstration of a hybrid framework for high-level road scene understanding, which integrates segmented road markings (T-1b) and domain knowledge into a hierarchical, graph-based description of the scene (T-1a) [19].

- A demonstration of an image de-raining framework to restore the road marking segmentation, which is negatively affected by lens distortions caused by adherent raindrops (T-1b and T-3b) [22].

- A demonstration of a model-driven approach for road marking classification from the acquired binary segmentation using additional domain knowledge regarding road construction (T-1a and T-1b) [23].

The contributions are outlined in more detail in the respective chapters.

1.4 Thesis Outline

This thesis is split into eight chapters. Following the introduction in this chapter, Chapter 2 introduces fundamental representations and data principles used in road scene understanding tasks and this thesis.

Chapter 3 improves the scalability and robustness of semantic segmentation under wide-ranging environmental conditions. Adverse weather and lighting decrease the performance drastically when the input condition is not equal to the training condition. DNNs are traditionally trained for every condition separately, but we demonstrate that this is suboptimal. Firstly, pixel-wise labels have to be acquired for every respective condition, which does not scale well. The required labelling effort is reduced significantly by employing image-to-image translation

techniques to generate vast quantities of training pairs for different conditions efficiently. Moreover, we demonstrate a new network architecture consisting of lightweight input adapters to learn an appearance-invariant representation optimal for semantic segmentation explicitly.

As discussed previously, the pixel-wise output representation obtained by deep semantic segmentation networks is limited in its direct usefulness for planning and decision making. The remaining chapters of this thesis aim to resolve some of these limitations.

Chapter 4 demonstrates a graph-based road layout representation, the *scene graph*, which can directly feed into planning and decision-making algorithms. This approach builds a bottom-up understanding of the road layout in a bird's-eye view from segmented entities (i.e. road markings and curbs) and is able to leverage prior domain knowledge regarding road construction. Chapters 5 - 7 present methods to obtain the necessary prerequisites for this representation efficiently.

Chapter 5 obtains a binary road marking segmentation as a step towards semantic classification. In order to achieve this efficiently, an additional sensor modality and domain knowledge are leveraged to generate training pairs automatically. Furthermore, we demonstrate that lens distortions (i.e. adherent raindrops) significantly degrade the segmentation performance. This issue is resolved by training a de-raining model to restore the image and thereby the segmentation.

Chapter 6 compares a model-driven and data-driven approach for classifying the road markings semantically by employing the binary road marking segmentation. The model-driven approach leverages additional domain knowledge regarding road construction to fit road marking models to the binary segmentation. The data-driven approach uses state-of-the-art image-to-image translation techniques to synthesize photo-realistic images according to predefined labels, thereby generating vast quantities of the desired training pairs automatically.

Chapter 7 improves IPM, which is another prerequisite for generating the scene graphs since this view is more convenient for reasoning about the road layout and explicitly encodes distances, making it more closely linked to the vehicle's

action space. However, the traditional homography-based transformation deforms the shape of the road markings, especially those farther away, which may lead to substantial errors in the semantic interpretation of scenes. We use a novel network architecture to overcome this limitation which generates a learned IPM, called *boosted IPM*, in a self-supervised way. Its benefits are demonstrated in virtual and real-world situations.

Chapter 8 provides a summary of the work presented in this thesis, a discussion on the broader impact of our work, and ideas for future work.

1.5 Publications

This thesis includes the following publications:

> [17] H. Porav, **T. Bruls**, and P. Newman, "Don't worry about the weather: Unsupervised condition-dependent domain adaptation", in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Oct. 2019, pp. 33-40. (Chapter 3)
>
> [19] L. Kunze, **T. Bruls**, T. Suleymanov, and P. Newman, "Reading between the lanes: Road layout reconstruction from partially segmented scenes", in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Nov. 2018, pp. 401-408. (Chapter 4; Appendix A)
>
> [18] **T. Bruls**, W. Maddern, A. A. Morye, and P. Newman, "Mark yourself: Road marking segmentation via weakly-supervised annotations from multimodal data", in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 1863–1870. (Chapter 5)
>
> [22] H. Porav, **T. Bruls**, and P. Newman, "I can see clearly now: Image restoration via de-raining", in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 7087-7093. (Chapter 5; Appendix B)

[23] P. Amayo, **T. Bruls**, and P. Newman, "Semantic classification of road markings from geometric primitives", in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Nov. 2018, pp. 387-393. (Chapter 6; Appendix C)

[20] **T. Bruls**, H. Porav, L. Kunze, and P. Newman, "Generating all the roads to Rome: Road layout randomization for improved road marking segmentation", in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Oct. 2019, pp. 831-838. (Chapter 6)

[21] **T. Bruls***, H. Porav*, L. Kunze, and P. Newman, "The right (angled) perspective: Improving the understanding of road scenes using boosted inverse perspective mapping", in *Proceedings of the Intelligent Vehicles Symposium (IV)*, Jun. 2019, pp. 302-309. (Chapter 7)

2

Representations & Data for Road Scene Understanding

Contents

2.1 Representations for Road Scene Understanding 13
2.1.1 Effective Representations 14
2.1.2 Non-Parametric Representations 15
2.1.3 Hybrid Representations 17
2.2 Data for Road Scene Understanding 19
2.2.1 Manually-Labelled Datasets 20
2.2.2 Physics-Based Simulators 20
2.2.3 Unlabelled Datasets 21

This chapter provides a broad overview of the representations and data principles used for road scene understanding, focussing on state-of-the-art methods. Specific literature studies are provided within the reproduced publications in Chapter 3 - 7 (and Appendix A - C) and are not repeated here.

2.1 Representations for Road Scene Understanding

This section introduces the principles of effective representations for road scene understanding. Subsequently, we review non-parametric and hybrid representations

to demonstrate that improvements are necessary for real-world deployment, some of which are presented in this thesis.

2.1.1 Effective Representations

The principles of effective representations for road scene understanding are grouped below in four categories: decision making, cost-effectiveness, versatility, and robustness.

For safe decision making, effective representations ideally:

- include the *road rules* conveyed by the road markings painted on the road surface. These provide an understanding of specific *road layouts* instead of road topologies.
- encode the *distances* of objects in the scene explicitly.
- reason about the *uncertainties* stemming from perception measurements.
- offer an abstraction level which directly *links with the vehicle's action space* (i.e. steer, stop, and go) and consequently can provide intuitive *explanations*.

In order to ensure cost-effectiveness, effective representations ideally:

- employ *self-supervised* learning frameworks to limit the required amount of expensive manual labelling.
- leverage *domain knowledge* to limit the required amount of data and computational power.
- are obtained with a *minimal sensor suite*, preferably solely containing imaging cameras as these are relatively inexpensive.

In order to remain versatile, effective representations are ideally:

- obtained from *in-situ perception* and do not require aerial images or aggregation of future data.

- extendable to *a wide variety of traffic scenes under various conditions and viewpoints* without fine-tuning.

In order to ensure robustness, effective representations are ideally:

- *invariant to appearance changes* caused by adverse weather conditions or illumination.
- *invariant to occlusions* caused by objects in the scene.
- *invariant to degradation* caused by wear.

The representations which we devise in Chapter 3 - 7 aim to follow these principles.

2.1.2 Non-Parametric Representations

State-of-the-art non-parametric representations for describing road scenes (in the front-facing camera) generally train DNNs to perform pixel-wise semantic segmentation [10], [11]. Semantic segmentation is the task of classifying every pixel of an image by its semantic class label (e.g. road, sidewalk, or car). In this way, it is similar to image classification but at a pixel level. The advantage of these representations is that they are able to represent a wide variety of scenes because they are not limited by underlying hand-crafted parametric models. Semantic segmentation of road scenes in the front-facing camera has been studied rigorously [24], [25]. Nevertheless, in line with the main focus of this thesis, we limit our discussion here to non-parametric representations that describe the road layout in a bird's-eye view.

In [26], the authors retrieve a semantic bird's-eye-view map by jointly optimizing a CRF in the front-facing perspective and the bird's-eye view. Because they use a homography transformation, objects that are not on the road surface are represented unnaturally in the map, even after aggregation over multiple frames. More recent approaches use an encoder-decoder architecture to predict a semantic occupancy grid either from a monocular image [27], [28] or a stack of surround-view images

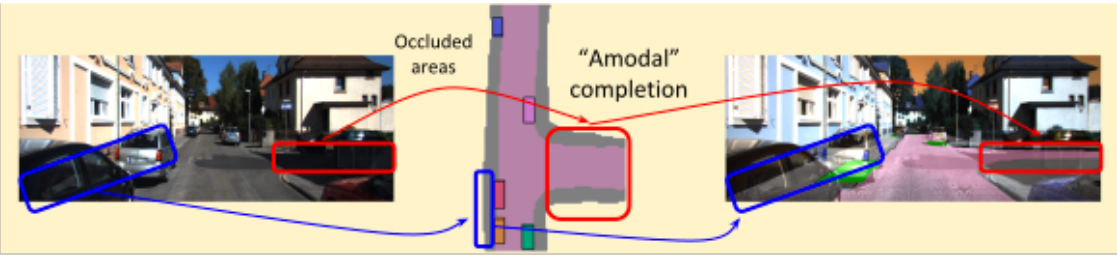


Figure 2.1: The latest non-parametric approach infers the road topology even in the presence of occlusions from a single image [35]. However, these approaches fail to capture the lane geometries, which are necessary for navigating through complex environments.

[29]. However, this architecture leads to a loss of spatial context resulting in coarse, non-sharp output predictions with inaccurate shapes and distances, especially for smaller classes. This issue is resolved in [30] by incorporating a dense transformer layer which preserves more spatial information and in [31] by "lifting" the image into a frustum of features before "splatting" them into a rasterized bird's-eye-view grid.

Another disadvantage of these methods is that they perform poorly in the presence of occlusions. Several works have proposed solutions for this issue. The authors of [32] align simulated semantic bird's-eye-view images, in which objects and occluded areas are accurately represented, with real-world data to indicate unobserved areas. Similarly, the vehicle's footprint is predicted in a bird's-eye view to prevent the stretching of objects that are not on the road surface in [33]. The authors of [34] presented an approach that reasons about the road topology in the bird's-eye view even if dynamic objects occlude the scene. They leverage monocular depth estimation and semantic segmentation as well as prior domain knowledge stored in OpenStreetMap to achieve this. The latest research [35], [36] predicts the road topology outside of the field-of-view and behind occlusions, as illustrated in Figure 2.1, without requiring depth estimation or semantic segmentation by leveraging adversarial feature learning. However, the output representation of these models describes only the road topology and not the road layout including the lane geometries, which is necessary for navigating through complex urban environments.

2.1.3 Hybrid Representations

The major disadvantage of the non-parametric approaches is that the pixel-wise representation fails to provide a direct and interpretable link with the vehicle's action space. Hybrid representations have been devised to resolve this limitation by leveraging the versatility of learned features or segmented entities of DNNs in combination with parametric models to describe the road layout. These hybrid representations can be clustered into three categories: coarse road representations, fine-grained road layout representations, and graph-based representations.

Coarse Road Representations

One of the earliest approaches [37] estimated the intersection topology and pedestrian crossings based on semantic segmentation features. The authors of [38] similarly employ DNN features to predict coarse scene attributes from images. Notably, they train their DNN in a self-supervised way by retrieving these attributes from OpenStreetMap. However, the representations of both are too coarse to describe complex road layouts.

A more detailed model, which is able to describe the road topology of complex intersections, is proposed in [39]. Nevertheless, the method fails to provide online predictions because multiple modalities such as vehicle tracklets, vanishing points, and scene flow are required. Furthermore, the model is unable to describe the lane geometries of complex multi-lane scenes, which is necessary for decision making. A similar model is used in other works to estimate intersections [40] and refine semantic segmentation [41].

In [42], semantic segmentation obtained with a DNN is combined with a parametric model to enhance existing maps with detailed semantics such as lanes, parking spots, and sidewalks. This model reasons about different road layouts but requires aerial images as an additional input.

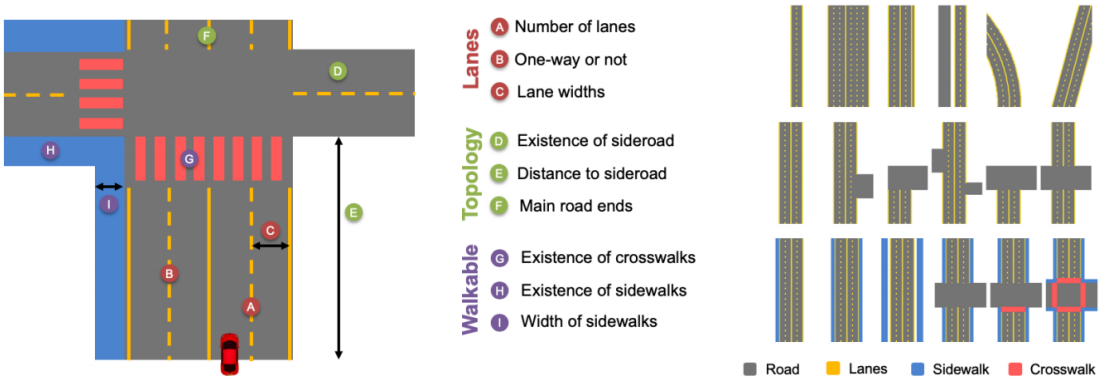


Figure 2.2: The parametric representation introduced in [43] describes the road topology as well as the road layout and thus allows for high-level reasoning in complex real-world scenarios.

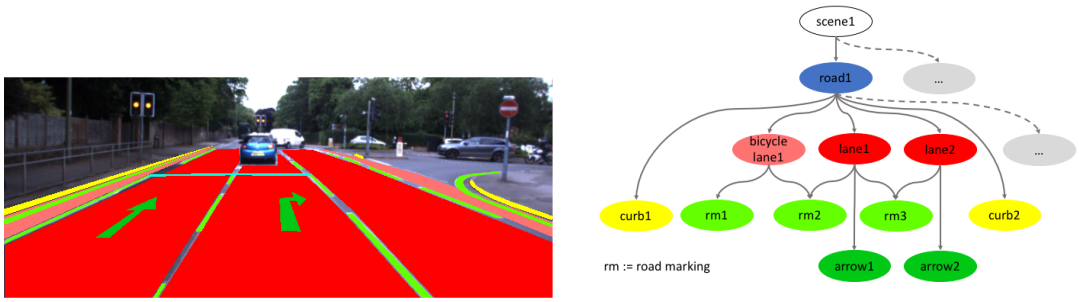
Fine-Grained Road Layout Representations

The development of fine-grained models that provide an accurate representation of the road layout was mainly restricted by the lack of adequate data. This changed when [43] introduced a new dataset along with a novel parametric top-view model, which allows for high-level reasoning to describe complex road scenes, as visualised in Figure 2.2. Semantic segmentation acquired with a DNN is used to predict scene parameters, which are then refined with an energy-based optimization. The same authors improved upon this initial work recently by leveraging camera motions, including context cues such as vehicles, and incorporating long-term video information [44].

Graph-Based Representations

None of the aforementioned approaches leverage the fact that road scenes are built hierarchically. Low-level cues such as road markings and curbs, which DNNs can accurately segment, can be grouped to form lanes, road layouts, and ultimately describe the entire scene. This principle, as shown in Figure 2.3, is exploited in several works.

In [45], hierarchical graphs fuse local information from different sensor modalities with prior and contextual information to model traffic scenes. Nevertheless, the output is still a pixel-wise representation.



(a) **(b)**

Figure 2.3: Road scenes are built hierarchically, **(a)**. This thesis centres around a graph-based representation, **(b)**, which describes these hierarchies starting from low-level cues such as road markings and curbs to form lanes, road layouts, and ultimately road scenes.

The authors of [46] introduce a theoretical hierarchical framework including uncertainties to reason about multiple hypotheses for the lane geometries. Similar methods that work on real-world data are introduced in [47]. A graph is built from linear patches of lane markings according to their spatial relationships, represented by continuous distributions, and non-parametric belief propagation is used to infer the road layout. However, these methods are not guaranteed to work in urban environments with complex intersections.

In [48], the lane separators are modelled as latent variables without linear constraints so that the framework becomes applicable to more complex scenes. By encoding geometric relationships at different levels (i.e. lane markings, lane separators, lanes, and the road), the authors show that they improve inference of the lane geometries even in case of false detections at the root nodes. This thesis centres around a similar approach, introduced in Chapter 4, which integrates segmented road markings and curbs into a graph-based representation. The latest research [49] incorporates graph-based reasoning directly into the DNN to infer the behaviour of vehicles in the scene while observing simple road layouts.

2.2 Data for Road Scene Understanding

Datasets for urban scene understanding can be categorised into three groups: labelled real-world datasets, physics-based simulators (which are labelled by design),

and unlabelled real-world datasets. In the first category, large-scale datasets exist for scene understanding tasks such as 3D object detection and tracking, VO, and scene flow estimation using extensive sensor suites [50]–[52]. Nevertheless, in line with the main focus of this thesis, we limit our discussion here to datasets for road scene understanding and focus on the ones used in this thesis.

2.2.1 Manually-Labelled Datasets

Progress in road scene understanding has depended significantly on the availability of (pixel-wise) labelled datasets [7]. The Cityscapes dataset [53] has become the de-facto benchmark, and we thus employ models pretrained on this dataset in Chapter 3 and 6. Nevertheless, the environments and conditions in Cityscapes are homogeneous, and the number of labels is still relatively low. Several larger, more heterogeneous datasets have been released subsequently such as the Mapillary Vistas dataset [13], the Audi Autonomous Driving dataset [54], and BDD100k [55]. The latter is used in Chapter 3 for evaluation. The authors of [56] have recently published the first dataset for high-level scene understanding beyond pixel-wise semantic segmentation.

Crucially, few of these datasets contain (semantic) road marking labels. A single road marking class is included in CamVid [12], Rainy RobotCar [22], and Mapillary Vistas [13], but this is not sufficient for understanding the driving directions they convey. Nevertheless, we use the former two to evaluate the presented road marking segmentation frameworks in Chapter 5. The Audi Autonomous Driving dataset [54] coarsely clusters different road marking types such as separators, lane boundaries, and symbols. Only the ApolloScape dataset [14] and VPGNet [15] include a wide variety of classes based on their underlying meaning. The differences are visualised in Figure 2.4. However, it remains extremely expensive to extend such datasets to all environments and conditions encountered during real-world deployment.

2.2.2 Physics-Based Simulators

As an alternative to manual labelling, gaming engines are used as virtual environments to test autonomous driving and collect (labelled) data on a large

(a) **(b)** **(c)**

Figure 2.4: Pixel-wise road marking labels in manually-labelled datasets are available in three different variants: a single road marking class as in CamVid [12], **(a)**, several coarse classes (e.g. separators, symbols, and boundaries) as in the Audi Autonomous Driving dataset [54], **(b)**, or all classes differentiated by their semantic meaning as in the ApolloScape dataset [14], **(c)**.

(a) **(b)**

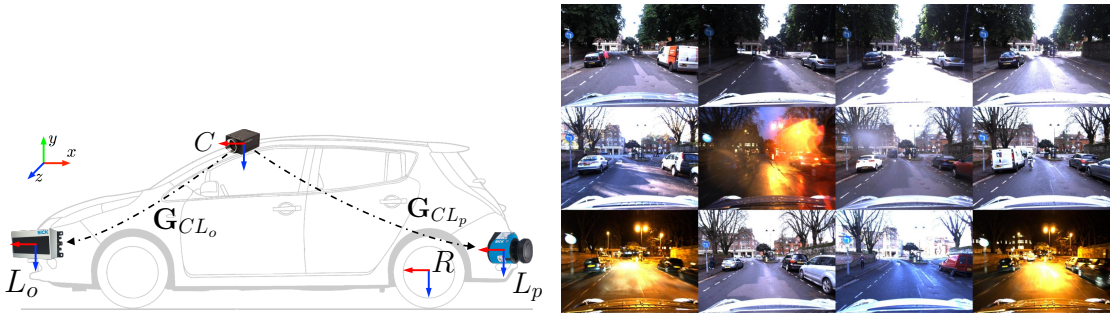
Figure 2.5: An example of a bird's-eye-view training pair in the CARLA simulator. The camera position is adjusted to retrieve a ground-truth bird's-eye view (displayed at a 90° rotation), **(b)**, aligned with the front-facing camera image, **(a)**.

scale. Some examples are SYNTHIA [57], Grand Theft Auto V [58], and AirSim [59]. They provide the ability to design and collect the desired data but are expensive to create and maintain. They also often lack the fidelity of the real world, thereby introducing a domain gap.

We use the CARLA simulator [60] and adjust the camera position to retrieve a perfect bird's-eye view aligned with the front-facing camera image, as shown in Figure 2.5. This dataset is used to evaluate the boosted IPM framework quantitatively in Chapter 7.

2.2.3 Unlabelled Datasets

Many urban driving datasets lack detailed semantic labels because of the high production costs. Besides, autonomous vehicles frequently encounter new situations and environments that do not exist in the labelled datasets. Therefore, there is a need to design self-supervised frameworks that are able to work with unlabelled data.



(a) **(b)**

Figure 2.6: The Oxford RobotCar platform and dataset. We use the data collected by three sensors attached to the platform (i.e. the front-facing camera, the front-facing LiDAR, and the rear-mounted pushbroom LiDAR), **(a)**, for road scene understanding. The dataset consists of multiple traversals of the same route under a wide range of environmental conditions, **(b)**.

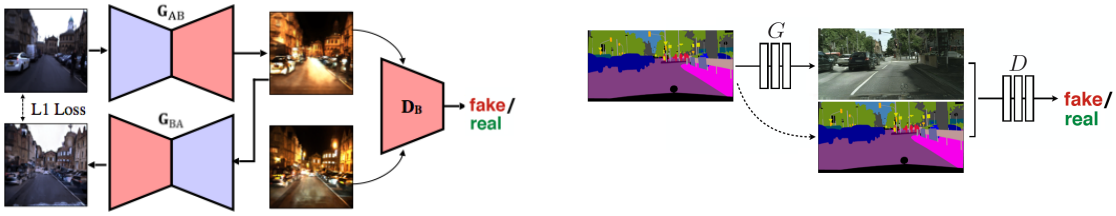
Oxford RobotCar Dataset

The Oxford RobotCar dataset [61] is used as the primary dataset in this thesis. It consists of more than 100 repetitions of a 10-kilometre route through urban environments and is captured under a wide range of weather, lighting, and traffic conditions. The data was collected while manually driving a Nissan LEAF equipped with an extensive sensor suite. In this thesis, we exclusively use the front-facing Point Grey Bumblebee XB3 colour stereo camera during deployment. Additionally, we leverage two SICK LMS-151 2D LIDARs to generate training data offline: one mounted in push-broom configuration at the back and one vertically attached to the front. This is shown in Figure 2.6.

We built upon a long legacy of existing research within ORI (formerly MRG) for LiDAR-camera calibration [62], visual localisation [63], visual navigation and mapping [64], visual odometry [65], amongst others.

Automatically Generating Training Pairs

We provide some examples of the two common paradigms employed in this thesis for generating training pairs automatically: generating a semantic label for a captured image or synthesizing a photo-realistic image for a predefined semantic label.



(a) (b)

Figure 2.7: Two ways in which image-to-image translation networks can be employed to generate training pairs automatically. In Chapter 3, we use a cycle-consistency GAN to change the appearance of an image, **(a)**. In Chapter 6, we use a conditional GAN [68] to synthesize photo-realistic images from altered semantic labels, **(b)**.

A semantic label is usually generated automatically for an image by bootstrapping different sensor modalities or incorporating domain knowledge. For instance, the approximated road marking labels are generated by leveraging LiDARs and domain knowledge in Chapter 5. Another example is driveable path planning [66], [67], which is achieved by projecting the future driven path into the image to generate a label.

Synthesizing new photo-realistic images for predefined labels is possible in several ways. Firstly, new viewpoints can be generated from a coloured point cloud, as in [69], or VO can be employed to stitch bird's-eye-view labels, as in Chapter 7. Secondly, the appearance of the image can be altered by using domain knowledge to model conditions such as rain, as in Appendix B, or fog [70]. However, it is difficult to model all minor details of appearance changes realistically, and therefore we use a cycle-consistency network for this purpose in Chapter 3. A third option is to modify the scene by adjusting the semantic map, as in [71]–[73] and Chapter 6, and subsequently synthesize a new corresponding image with an image-to-image translation network. The latter two of these options are illustrated schematically in Figure 2.7.

3

Self-Supervised Scene Segmentation under Wide-Ranging Environmental Conditions

Contents

3.1 Publication **27**
3.2 Summary of the Results **36**
3.3 Conclusion **37**

This chapter extends the general case of semantic segmentation under overcast conditions towards a wide range of environmental conditions. When the input condition is substantially different from the condition for which the DNN was trained (generally overcast), the performance decreases drastically [74]. Parts of the image might also be occluded (e.g. overexposed, dark, or covered by raindrops), making it difficult to reason about these regions. Traditionally, separate condition-dependent DNNs [70], [75] are trained and selected according to the current input condition. Nevertheless, we demonstrate that this is suboptimal in terms of scalability and robustness in the reproduced publication in Section 3.1.

Condition-dependent DNNs require pixel-wise labelled data for every respective condition. Consequently, extending these towards all conditions increases the required manual labelling effort drastically and does not scale well. We reduce the

labelling effort significantly by employing state-of-the-art image-to-image translation techniques to change the condition of the training images. Moreover, a lightweight, expandable framework of condition-dependent input adapters is implemented to "strip" the condition of the input image and thereby improve robustness under appearance changes. By splitting the stripping and output tasks, we learn general semantic representations across and independent of the input conditions.

Both improvements leverage the fact that the ground-truth semantic segmentation of the scene is independent of the appearance of the image, i.e. the same scene during the day and night results in an equivalent semantic segmentation – for static classes and infrastructure. We pair the semantic labels of the reference condition (i.e. overcast) with images of various conditions by changing only their appearance using multiple cycle-consistency GANs. This technique significantly reduces the labelling effort and has now gained more traction [76]. Additionally, the fact that the semantic segmentation of a scene is invariant to the input condition acts as the intuition behind the proposed network architecture. We assume that an appearance-invariant representation exists in which the environmental condition is "stripped" from the image, which is the optimal input for the pretrained semantic segmentation network. Lightweight, condition-dependent input adapters are trained to map the input images of every respective condition towards this representation by using the pretrained network as a supervisor. We demonstrate that this approach is superior to training separate condition-dependent DNNs because we explicitly direct the network to learn a useful mid-level representation.

In summary, this chapter makes the following principal contribution:

- The concept of underlying appearance-invariant representations, which comprise all necessary information for decision making and other relevant tasks in autonomous driving (T-3a).

Additionally, this chapter makes the following supporting contributions in collaboration:

- A demonstration of an image-to-image translation technique that generates semantic training data for wide-ranging environmental conditions by changing the appearance of the image, thereby significantly reducing the required labelling effort (T-2c).
- A demonstration of a lightweight, expandable framework that explicitly learns an appearance-invariant representation optimized for semantic segmentation (T-3a).

3.1 Publication

This section contains a reproduction of the following publication:

> [17] H. Porav, **T. Bruls**, and P. Newman, "Don't worry about the weather: Unsupervised condition-dependent domain adaptation", in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Oct. 2019, pp. 33-40.

Don't Worry About the Weather: Unsupervised Condition-Dependent Domain Adaptation

Horia Porav, Tom Bruls and Paul Newman

***Abstract*— Modern models that perform system-critical tasks such as segmentation and localization exhibit good performance and robustness under ideal conditions (i.e. daytime, overcast) but performance degrades quickly and often catastrophically when input conditions change. In this work, we present a domain adaptation system that uses light-weight input adapters to pre-processes input images, irrespective of their appearance, in a way that makes them compatible with off-the-shelf computer vision tasks that are trained only on inputs with ideal conditions. No fine-tuning is performed on the off-the-shelf models, and the system is capable of incrementally training new input adapters in a self-supervised fashion, using the computer vision tasks as supervisors, when the input domain differs significantly from previously seen domains. We report large improvements in semantic segmentation and topological localization performance on two popular datasets, RobotCar and BDD.**

I. Introduction

Robust Computer Vision is paramount to the prevalence of general-purpose robotics, and even more so in fields such as autonomous transportation, where failures may be catastrophic. Modern models that perform system-critical tasks such as segmentation, detection, localization and classification - either traditional heuristics-based or learned - exhibit good performance and robustness under ideal conditions (i.e. daytime, overcast) but performance degrades quickly and often catastrophically when input conditions change. To have their breakthrough, real-world systems must work under varying illumination, weather and noise conditions, and in the long term will need the ability to adapt to new, unseen domains without explicit supervision.

As a type of domain adaptation technique, domain unification is the holy grail of visual perception, theoretically allowing models trained on samples with limited heterogeneity to perform adequately on scenes that are well out of the distribution of the training data. Domain unification can be applied within the vast distribution of natural images [1], [2], [3], between natural and synthetic images (computer-generated, whether through traditional 3D rendering or more modern GAN-based techniques) [4], [5] and even between different sensor modalities [6]. Additionally, domain unification can be implemented at different stages of a computer vision pipeline, ranging from direct approaches such as domain confusion [7], [8], [9], fine-tuning models on target domains [1] or mixture-of-expert approaches [10], etc.

However, a major limiting factor for all these approaches is the scarcity of labelled multi-modal data, driven by the high cost of manual labelling. Most approaches attempt to solve this shortcoming by using 3D-rendered simulations that programatically provide ground-truth [11], [12], or by using unsupervised techniques that adapt models based on auxiliary or proxy tasks [7], [8], [9], [13], [1].

Authors are from the Oxford Robotics Institute, University of Oxford, UK. {horia, tombruls, pnewman}@robots.ox.ac.uk

Fig. 1. Our method allows off-the-shelf models to work with new, unseen domains, without any specific fine-tuning. In this example, we use our input adapter to allow a segmentation model trained using only daytime examples to work under night-time conditions. Top-left quadrant is the input night-time image, top-right quadrant is the output of our input adapter. The bottom-left quadrant is the output of the segmentation model applied on the original night-time input image. The bottom-right quadrant is the output of the segmentation model applied on the output of our adapter. The initial segmentation result is unusable, while the result obtained by running the model on the output of our domain adaptation pipeline accurately classifies roads, pavement, pedestrians, bicycles, vegetation and buildings.

We propose a hybrid method, where multi-modal data is generated in an unsupervised fashion with approximated, high-quality ground truth, followed by supervised training of domain-adapters, for a battery of computer vision tasks, using this generated data and approximated ground truth. While this final step is supervised, the data used for supervision is itself created in an unsupervised fashion, making the entire pipeline unsupervised. To do so, we start by generating multi-modal training data: from a database of image sequences categorized using the time of day and weather conditions at their moment of recording, we select a daytime, overcast, clear **reference** sequence. We leverage the fact that modern computer-vision pipelines perform excellent (e.g. $\geqslant$ 0.83 mIOU on the Cityscapes multi-class segmentation validation set [14]) on inputs with ideal conditions, and thus we run this **reference** sequence through a set of off-the-shelf computer vision tasks and save the outputs as *approximated* ground truth - these results are not 100% identical to the real-world ground truth but they approximate it very well.

Secondly, we apply style transfer to the **reference** sequence in order to produce a set of sequences which are structurally and geometrically identical to the **reference** se-

quence but differ in appearance. The applied style is sourced from many other sequences, each possessing a different appearance. This step is key to our approach - retaining the structure and geometry of the **reference** sequence while varying appearance means that the *approximated* ground truth data is still valid: as an example, the same car but with varying appearances - once during daytime and once during nighttime - will still have the same ground truth footprint in a semantic segmentation map. At the end of these two steps we will have produced - in an unsupervised fashion - a set of sequences with varying appearances accompanied by task-specific ground truth data.

In step three, instead of training condition-dependent, separate task-specific models (e.g. a segmentation model for day-time, one for night-time etc.), we train lightweight condition-specific image adapters that are then used with vanilla, off-the-shelf task-specific models. The motivation for this is simple, but important: models that are invariant to input distributions are notoriously difficult to architect, parametrize and learn, while multi-model approaches such as mixture-of-experts do not scale well with the variance of the input distributions due to memory and runtime constraints. We tackle both these problems by training small, lightweight condition-specific convolutional input adapters, while a classifier-supervisor chooses the best adapter to be run, dependent on the distribution of the inputs. This approach adds minimal overhead to any off-the shelf task, benefits from parameter-counts that do not depend on the variances of the input distributions, and lends itself well to online learning of new conditions. Additionally, in contrast to the larger, task-specific models, the image adapter models tend to take up very little storage space and runtime memory and can be nearly-instantaneously loaded and re-loaded by the processing pipeline.

The final stage of our approach allows a robot or vehicle to incrementally adapt to a new, unseen domain: if the condition of the input images does not match one that the system has been previously trained on, the unsupervised style transfer pipeline will select a model that is closest to the current condition, clone it, and fine-tune this cloned model to be able to change the style of the **reference** sequence so that it matches the style of the current input images. Afterwards, data generated using this new model will be used to train - in a supervised fashion - an additional condition-specific image adapter that will allow upstream computer vision tasks to perform well on the new input image condition.

We benchmark our approach on two important tasks in computer vision and robotics: semantic segmentation and image retrieval/topological localization. This list is obviously not exhaustive, and the addition of extra supervisory tasks may lead to further improvements in performance.

Our main contributions include:

- Using cycle-consistency GANs to generate multi-condition training data with approximated ground truth for a battery of off-the-shelf computer vision tasks.
- Training input image adaptors by using the off-the-shelf computer vision models to generate a supervisory signal.
- Enabling online learning of new, unseen domains by leveraging the unsupervised data generation pipeline along with domains on which the data generation models have already been trained.
- Showing that training multiple lightweight adapter modules is better than training monolithic computer vision models that are invariant to input distributions.

Our qualitative and quantitative results are presented in section V.

II. Related Work

A. Computer Vision Tasks

Semantic Segmentation: Semantic segmentation is a key task in robotics, and modern approaches exhibit very good performance when input conditions are favourable. Deep convolutional models such as Deeplab V3+ [14], SDN [15] or PSPNet [16] achieve high class-mIOU figures ($\geqslant 80\%$) on benchmarks such as Cityscapes [17], but their performance breaks down fast when their inputs change due to different weather conditions, seasons or times of day. For this work, we chose DeepLab V3+ as the reference model due to its excellent open-source implementation and availability of results on a number of popular benchmarks.

Topological Localization: Similarly, widely used topological localization frameworks such as FABMAP [18], SeqSlam [19] or NetVLAD [20] achieve high recall and precision figures on clear, daytime images or when explicitly matching images with the same condition (e.g. winter-winter matching), but break down when the conditions of the locations to be matched differ. For this work, we chose NetVLAD as the reference topological localizer.

B. Domain Adaptation

Domain Confusion: The most common approaches fall under the umbrella of domain confusion, making use of a discriminator that forces features extracted by an encoder to follow a similar distribution for both a source and a target domain [7], [8], [9], [13], [21], [22]. The downside of these approaches is the lack of a direct loss for the target domain, which limits its upper bound on performance.

Style-Transfer: Other approaches attempt to directly train computer vision models using synthetic data generated via style-transfer, or to directly adapt the input data to the target domain. Notable approaches include those of [4], [5], [23], [3] and [2]. Generally, these methods seem to have the most promise of reducing the domain gap between real and synthetic images, hence our decision to generate training data using the approach of [24].

In [25], a style-transfer pipeline is trained incrementally by feeding the segmentation map, obtained at a previous adaptation step, as an auxiliary input at each incremental step. The downside is that this type of self-supervised approach assumes that high values in the softmax layer (using segmentation as an example) automatically correlate with higher prediction accuracies, whereas we only approximate ground truth labels once, on a reference, high-quality input sequence using models whose accuracies have actually been validated experimentally.

The closest to our approach is [1], where a semantic segmentation network is trained first with a day-time hand-labelled dataset, and then used to predict labels on intermediary datasets recorded at incremental types of twilight. The

twilight images and estimated labels are then used to further fine-tune the segmentation model, which is finally used to segment night-time images. In contrast, our approach only computes approximated labels on the **reference** condition and uses the rest of the conditions as a guide for style transfer. However, the approach of [1] could be used as a drop-in replacement for style transfer in the larger context of our framework. Similarly, [26] trains segmentation models with a mixture of source domain images and synthetic images with the style of an incrementally-shifted target domain. The main difference between these approaches and our work is that instead of directly training or fine-tuning computer-vision tasks, we train lightweight input adapters while using the performance (loss) on these tasks as a supervisory signal.

Additionally, [27] presents an approach where an encoder-decoder is trained to transform the appearance of input images to a reference appearance, but this is only benchmarked on scenes with small changes in appearance, in the context of 6-DOF localization.

C. Online Learning

The authors of [28] present an approach to incremental online domain adaptation, making use of unsupervised training by employing domain confusion at the level of encoder features from both target and source domains, while slowly shifting both domains through a range of incremental appearance changes (e.g. day to night). In our case, the unsupervised regimen is moved to the data-creation stage, with the generated data being used for supervised training of the input adapters, leading to better training stability and better performing models. Additionally, the authors present a method of reducing data storage requirements by approximating the feature distribution of the source data (or reference data, in our case) using a generative model, which could potentially be swapped for the reference sequence in our approach.

Other approaches include map-management for lifelong learning [29] and adaptation on a domain manifold [30]. Our incremental learning pipeline follows the spirit of these works by always choosing to fine-tune a seed model that was initially trained on data from a domain that is close to the current target domain.

D. Expert Systems

Finally, systems [31], [10] exist that attempt to achieve invariance to the input conditions by running multiple sub-models in parallel and combining their outputs using a weighting or gating scheme to yield the desired result. One major issue with this type of approach is that runtime memory and processing power requirements increase linearly with the number of expert components used in parallel. While we also produce a number of input adapters that is proportional to the number of target domains, our classifier will choose only one input adapter to be run, per target domain, leading to a very small computational and memory footprint at runtime.

III. Learning Condition-Dependent Representations

A. Synthetic Multi-Condition Data

The first step in our approach is data generation. From the Oxford Robotcar Dataset [32], we select a **reference** sequence - one that is daytime, clear, overcast - and a number of traversals with difficult conditions - night, rain, snow etc. We use these conditions, along with a cycle-consistency architecture GAN [24], to train generative models that can apply style transfer to the **reference** condition in order to create a number of synthetic sequences that maintain the structure and geometry of the **reference** condition - locations, shapes and topologies of both static and dynamic objects and of the overall scene - but exhibit variation in appearance. In the following paragraphs we offer a succinct introduction to cycle-consistency GANs and how we use them to generate new data. The explanations offered here are in no way exhaustive, and interested readers should refer to the work of [24] for further details.

Following the work of [24], we employ 2 generators: given an image I_{A} from domain A (**reference**) and an image I_{B} from domain B (night, rain, snow etc.), we use generator G_{AB} to translate an image style from domain A to domain B and generator G_{BA} to translate an image style from domain B back into domain A. An adversarial loss is applied on the output of each generator: discriminator D_{B} on the output of generator G_{AB}, and discriminator D_{A} on the output of generator G_{BA}. The adversarial losses are formulated as:

$$\mathcal{L}_{\mathrm{B}_{\mathrm{adv}}} = (D_{\mathrm{B}}(G_{\mathrm{AB}}(I_{\mathrm{A}})) - 1)^2 \tag{1}$$

$$\mathcal{L}_{\mathrm{A}_{\mathrm{adv}}} = (D_{\mathrm{A}}(G_{\mathrm{BA}}(I_{\mathrm{B}})) - 1)^2 \tag{2}$$

The complete adversarial objective to be minimized $\mathcal{L}_{\mathrm{adv}}$ is:

$$\mathcal{L}_{\mathrm{adv}} = \mathcal{L}_{\mathrm{B}_{\mathrm{adv}}} + \mathcal{L}_{\mathrm{A}_{\mathrm{adv}}} \tag{3}$$

We train the discriminators to minimize the following objective:

$$\mathcal{L}_{\mathrm{B}_{\mathrm{disc}}} = (D_{\mathrm{B}}(I_{\mathrm{B}}) - 1)^2 + (D_{\mathrm{B}}(G_{\mathrm{AB}}(I_{\mathrm{A}})))^2 \tag{4}$$

$$\mathcal{L}_{\mathrm{A}_{\mathrm{disc}}} = (D_{\mathrm{A}}(I_{\mathrm{A}}) - 1)^2 + (D_{\mathrm{A}}(G_{\mathrm{BA}}(I_{\mathrm{B}})))^2 \tag{5}$$

The complete discriminator objective to be minimized $\mathcal{L}_{\mathrm{disc}}$ is:

$$\mathcal{L}_{\mathrm{disc}} = \mathcal{L}_{\mathrm{B}_{\mathrm{disc}}} + \mathcal{L}_{\mathrm{A}_{\mathrm{disc}}} \tag{6}$$

A cycle-consistency loss [24] is applied between the reconstructed and input images:

$$\mathcal{L}_{\mathrm{rec}} = \left\| I_{\mathrm{input}} - I_{\mathrm{reconstructed}} \right\|_1 \tag{7}$$

The final generator objective $\mathcal{L}_{\mathrm{gen}}$ is:

$$\mathcal{L}_{\mathrm{gen}} = \lambda_{\mathrm{rec}} * \mathcal{L}_{\mathrm{rec}} + \lambda_{\mathrm{adv}} * \mathcal{L}_{\mathrm{adv}} \tag{8}$$

with each λ term representing a hyperparameter that weighs the importance of each individual objective. We want to find the optimal generators G_{AB}, G_{BA} that minimize the complete objective:

$$G_{\mathrm{AB}}, G_{\mathrm{BA}} = \underset{G_{\mathrm{AB}}, G_{\mathrm{BA}}, D_{\mathrm{B}}, D_{\mathrm{A}}}{\arg\min} \mathcal{L}_{\mathrm{gen}} + \mathcal{L}_{\mathrm{disc}} \tag{9}$$

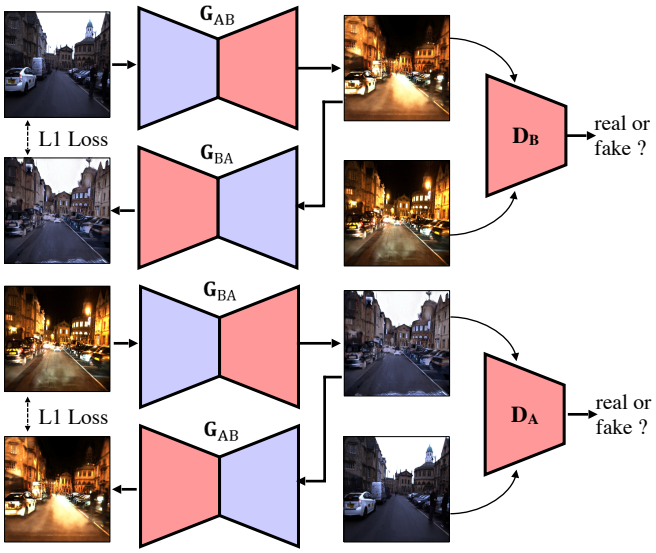


Fig. 2. The overall training architecture of the cycle-consistency GAN used to generate synthetic training data. We follow the training regimen described in [24] for each of the N pairings between the **reference** condition and a **target** condition.

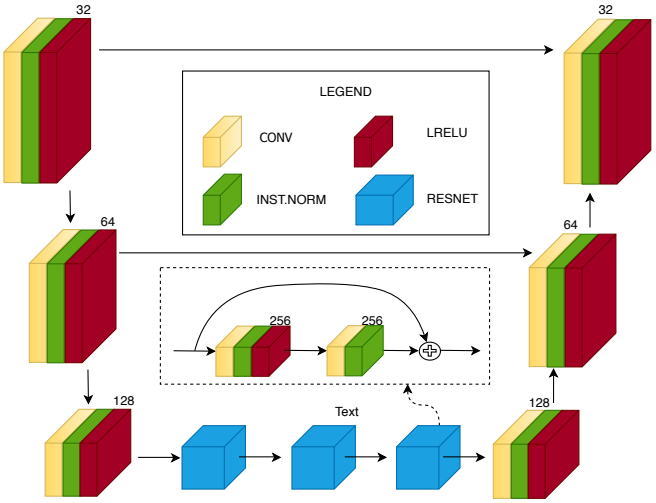


Fig. 3. The overall architecture of our input adapters. Under the assumption that a change in condition should not change the overall structure of the scene, we make use of skip-connections [34] and a ResNet [33] bottleneck to facilitate the direct transfer of features from the input side of the network to the output side.

We follow this methodology for N difficult conditions (domain B), always paired with the **reference** condition (domain A), yielding $2N$ generators. However, once the generators have converged, we only use the generators that apply the style of domains B to images from domain A - G_{AB} - to generate N versions of the **reference** sequence, each bearing the appearance of a sequence from domain B. Please note that for brevity we omit the condition-specific subscripts from the equations above. An overview of the CycleGAN architecture is shown if Figure 2.

B. Input Adapters

The second step in our approach is to use the data generated in the previous step to train a bank of adapters that preprocess the input images such that they follow a distribution similar to that of the training sets used to train the bank of tasks. We formulate our input adapters as convolutional encoder-decoders with 3 down-convolutions, a bottleneck with N_{res} ResNet [33] blocks and 3 transpose-convolutions. The input to our adaptors is a 3-channel RGB image, while the output is a 3-channel image compatible with the inputs of many well-known models (semantic segmentation, object detection, depth estimation etc). This configuration provides a light-weight solution that is easy to train using labelled data, with reduced storage requirements and a small run-time memory footprint.

For each input adapter F_{k} (specific to a k^{th} particular appearance), and each task T_m, we formulate the following loss:

$$\mathcal{L}_{T_m} = T_m(F_{\text{k}}(G_{AB_k}(I_A))) - T_m(I_A) \tag{10}$$

where G_{AB_k} is the CycleGAN generator that transforms images from the **reference** condition to the k^{th} condition, and I_A is an input **reference** image. For each input adapter F_{k} specific to condition k out of N conditions, and M tasks, the final objective to be optimized becomes:

$$F_{\text{k}} = \arg\min_{F_{\text{k}}} \sum_{m=1}^{M} \alpha_m * \mathcal{L}_{T_m} \tag{11}$$

where α_m is a non-negative weight modulating the importance of each task T_m.

One key takeaway here is that F_{k} is essentially different from G_{BA_k} (the domain-specific generator that maps an image back to the appearance of the **reference** sequence) - we are not directly concerned with obtaining images that possess the appearance of the reference sequence, instead we want to obtain a processed image that maximizes the performance on the set of tasks T_m. This can be observed in Figure 6, second image from left.

C. Domain Classifier

We employ a domain classifier D to select the most suitable input adapter F_k that enables optimal performance on input images with the k^{th} condition. The classifier follows a largely traditional architecture comprised of 4 convolutional layers and 3 fully connected layers followed by a softmax layer, outputting an N-length vector. Given an input image I_A and a domain label t as an N-length one-hot encoding, we wish to find the parameters of the classifier D that minimizes the cross-entropy between the output of the classifier and the target label t:

$$D = \arg\min_{D} -\sum_{k=1}^{N} t_k \cdot \log(D(I_A)_k) \tag{12}$$

with k used to denote the element in each one-hot encoding. After training the classifier with N conditions, we additionally use the output of the penultimate fully-connected layer as a length-128 condition descriptor, which allows us to discriminatively identify domains that are outside of the original N domains used during training. To do this, we average the descriptors over a sequence of input images, and compare to other stored descriptors in the Euclidean space. A detailed explanation of how this is used is presented in Subsection III-E.

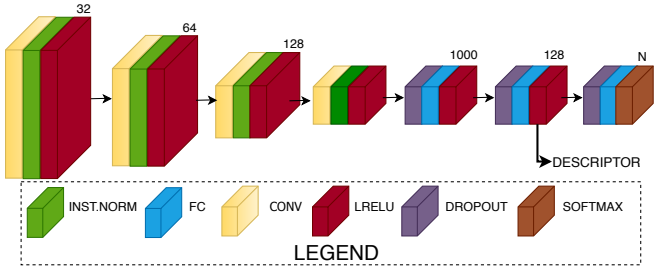


Fig. 4. Our classifier follows a traditional architecture, being composed of a series of down-convolutional layers followed by three fully connected layers. For on-line identification of new, unseen domains, we interpret the output of the penultimate fully-connected layer as a discriminative descriptor. Doing so allows us to identify an arbitrary number of domains, beyond the original N domains, without re-training the classifier.

D. Parameter Memory

After training each of the N condition-specific input adapters F_k parameters (weights) are stored in a memory (database) S. Additionally, the parameters of input adapters fine-tuned following the approach described in Subsection III-E are also stored in the same memory. The classifier described in Subsection III-C is used to select a set of optimal parameters to be used in the input adapter F_k. The memory can be queried in two ways: either by using an index k between 1 and N (the number of initial conditions) or by specifying a length-128 query descriptor and retrieving the set of parameters associated with the descriptor that is closest in the Euclidean space. To enable online learning of unseen domains, we additionally save the parameters of the cycle-consistency GAN generators using the same addressing scheme.

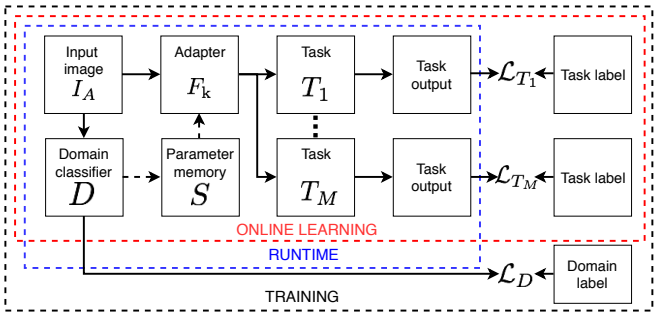


Fig. 5. An overview of our train- and test-time pipeline architecture. Given in input image I_A, the output of the classifier D is used to select a set of parameters for the input adapter F_k. The input adapter is then used to transform the input image into a representation that is better suited for the bank of M tasks T_m. During training, the performance of the tasks on the transformed input image is used as a corrective signal through the set of losses $\mathcal{L}_{T_m}$, along with a domain classification loss $\mathcal{L}_D$ used on the output of the classifier. At runtime only the components contained within the blue dotted rectangle are used. During online learning, the components contained within the red dotted rectangle are used. Solid arrow lines represent differentiable paths, while dotted arrow lines represent non-differentiable paths.

E. Online Learning

The pipeline described in the previous subsections can be extended to incremental, unsupervised, online learning of new, unseen domains without requiring any significant modifications to the existing system. We summarize and outline below the processed used:

- Given a continuous sequence of incoming images, we store the current frame and $T-1$ past frames in a buffer of length T that gets updated using a First-In-First-Out scheme
- For each frame in the buffer, we compute a length-128 condition descriptor using the penultimate layer of the classifier and average all the descriptors, yielding one single length-128 average descriptor
- If this average descriptor condition differs(in Euclidean space) by more than a threshold from the descriptors of any conditions previously trained on (i.e. the parameter memory S is unable to reliably identify the condition), the following training pipeline is triggered:
- We select the cycle-consistency GAN models closest to the current condition(using the condition descriptor), clone and fine-tune them for the current condition using the sequence stored in the buffer
- We use the newly trained generators from above to apply the new style to the **reference** condition to create a new training sequence
- We select the input adapter that is closest to the new condition(again using the descriptor), clone it and train it using the newly created training sequence from above
- We begin using this new adapter in the pipeline until the input condition changes significantly again

In the following section we describe our experimental setup.

IV. Experimental Setup

A. Creating multiple conditions

From the RobotCar Dataset [32], we choose $N = 7$ initial conditions: Snow, Dusk, Night, Night(rain), Night(low exp.), Shadows, Sun(glare) and a **reference** condition with a daytime, overcast condition. Additionally, we choose Sun (with ultrahigh exposure) as a condition not seen during initial training, to be used for online training.

B. Training

For training the cycle-consistency GAN models, we closely follow the approach from [24], and the reader is encouraged to consult the publication for further details. We train each of the initial N condition pairs for 100 epochs, and each online fine-tuning stage for 5 epochs. For training the input adapters, we use the Adam optimizer [35] with a base learning rate of 0.0005, $\beta_1 = 0.9$ and $\beta_2 = 0.999$. We have found that training an initial adapter on the **reference** condition (creating an identity function) and then using the parameters of this adapter as a 'seed' for training the initial N adapters greatly stabilizes and speeds up training. This provides the hint that most parametrisations for different condition adapters lie relatively close to each other on the parameter manifold, partly explaining the relative efficiency of our approach to incremental domain adaptation. We train the input adapters for 20 (offline) or 5 (online) epochs or until performance on the validation split stops increasing, whichever comes first.

C. The tasks

For our particular experiment, we chose $M = 2$ tasks: semantic segmentation and topological localisation, since they represent two critical components for robotics. For the semantic segmentation task, we chose to use DeepLab V3+,

as it produces state of the art results on a number of standard benchmarks [14]. We use a model checkpoint[1] trained on the Cityscapes dataset [17] that achieves 0.83% mIOU on the Cityscapes test split. For topological localisation, we chose the de-facto standard approach of computing place descriptors, NetVLAD [20], and used L2 matching. We use a model checkpoint[2] trained on the Pitts30K dataset [36].

Both of these architectures may be freely swapped with others as long as they are end-to-end differentiable. We set $\lambda = 1$ for semantic segmentation and $\lambda = 10$ for topological localisation, as we have noticed that increasing the importance of the localisation task improves overall performance without affecting the semantic segmentation task.

D. Performance

The input adapter performs inference at approximately 20 Hz for RGB inputs with a size of 640×480, on an Nvidia Titan V GPU. The chosen tasks have independent runtime performances of 3 Hz and 20 Hz for semantic segmentation and topological localisation, respectively.

We benchmark on-line learning by introducing an unseen condition (Sun with ultrahigh camera exposure). When starting from a system trained on $N = 7$ initial conditions described above, with the closest condition being Sun with glare, on-line training for the new domain takes approximately 30 minutes to first fine-tune the cycle-consistency GAN generators, followed by approximately 10 minutes to train the input adapter. This gives a complete cycle of 40 minutes, meaning that we can fine-tune for new domains approximately 36 times per day. This time should decrease with the addition of more conditions, as new domains could then benefit from 'closer' seeds when performing online training.

V. Results

For semantic segmentation, we create a testing split from the RobotCar Dataset [32] **reference** sequence and generate testing sequences with different conditions by applying style-transfer using the cycle-consistency GAN generators obtained during the training stage of our pipeline. This process yields sequences with 8 different conditions (the 7 initial conditions and one additional condition for testing online learning) and a common approximated ground truth. We again wish to remind the reader that testing is done on sequences derived from the **reference** sequence through style transfer (along with the approximated ground truth) due to a lack of semantic annotation for the RobotCar Dataset. To show the increase in segmentation performance on a dataset with hand-labelled groundtruth, we further test our system on the validation split of the BDD100K segmentation dataset [37]. As we train exclusively on data from the RobotCar Dataset, BDD is a domain that has *never* been seen during training of any components of our pipeline, and better reflects the usefulness of the proposed pipeline. As the BDD validation sequence does not have enough instances of each condition to also demonstrate online-learning, we freeze our system trained on 8 conditions ($N = 7$ initial conditions and 1 online-learned condition) and test it on BDD. For topological localisation, we test on **real** sequences (not produced using style-transfer) from the RobotCarDataset using the provided INS-RTK GPS ground truth, using a tolerance of 5 meters and reporting Precision-Recall and Area Under Curve (AUC).

A. Quantitative results

Semantic segmentation performance is significantly improved for RobotCar sequences, as can be seen in Table III. We compare our method (Indiv. adapters) of selecting input adapters with 3 other scenarios: a Baseline where the segmentation model is applied directly on the input image, one where we fine-tune the DeepLab segmentation model on all existing conditions (Deeplab-all) and one where we fine-tune individual DeepLab segmentation models for each condition, and use the classifier output to select the right model. The results show that our method consistently and significantly surpasses all other methods, with an average improvement of over 20 percentage points over the Baseline method. Additionally, night-time conditions show impressive gains in performance, with over 47 percentage points gained over Baseline for the **Night(rain)** condition. Table I presents results for semantic segmentation on the BDD dataset, which has never been seen during training. We test against the same 3 methods described above and again observe significant improvements, with over 5 percentage points gained in Mean Intersection Over Union compared to the Baseline method.

Similarly, topological localization shows a significant overall improvement, with an average of 10 percentage points of overall improvement in Area Under Curve (AUC), and very large improvements for Sun(glare) and Shadows. Night-time traversals are one exception where the improvements are still positive but smaller, as the task of detecting discriminative features is arguably harder than performing segmentation. Table II and Figures 8,7 present the results in more detail.

The condition classifier has an overall accuracy of 91%. The classifier confusion matrix is presented in Figure 9. The confusion of the reference, dusk and snow conditions does not lead, empirically, to a large drop in upstream task performance as there is a large degree of similarity between them.

B. Qualitative results

Additionally, we inspect segmentation results on **real** sequences (not produced using style-transfer) from the RobotCar Dataset. While they possess the same range of conditions as the ones the system was trained on, a ground truth is not available. We observe improvements in segmentation across the board, with the most important classes (vehicles, pedestrians, bicyclists etc) becoming distinguishable in even the most difficult conditions. An example for night-time is given in Figure 6.

C. Online learning results

To test our online learning capabilities, we run our system on a condition never before seen during training, **Sun(with ultrahigh camera exposure)**. The descriptor extracted from the penultimate layer of the classifier cannot be accurately matched to any stored descriptors, so the online training

[1] https://github.com/tensorflow/models/blob/master/research/deeplab/g3doc/model_zoo.md

[2] https://github.com/uzh-rpg/netvlad_tf_open

Fig. 6. Improvement of semantic segmentation on a **real** RobotCar night-time input. The first image is the input image, the second image is the output of the selected adapter, the third image is the result of running the segmentation model on the adapted image, while the last image is the result of running the segmentation model on the original, raw input image.

process is triggered. In the descriptor feature space, the closest condition stored is **Sun(glare)**, which is used as a seed for training the cycle-consistency GAN generators and a new input adapter. Results for this new condition are presented in Figures 7, 8 and in the *gray* shaded columns in Tables II and III. As with the initial condition, we observe a large and significant increase in performance for both topological localisation (over 15 percentage points) and semantic segmentation (11 percentage points).

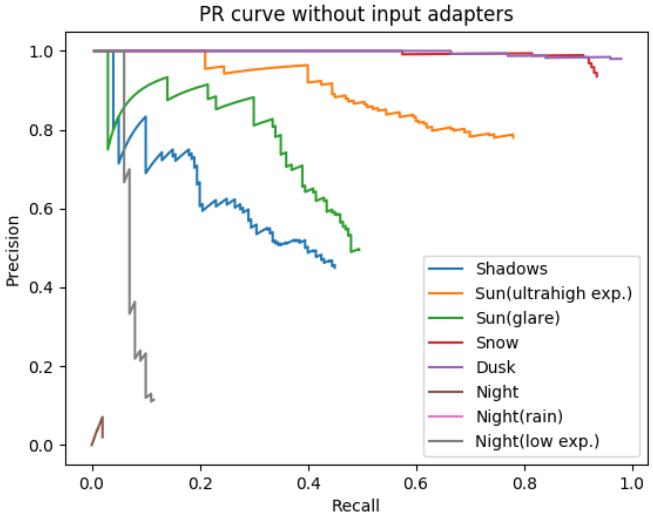


Fig. 7. RobotCar topological localisation Precision-Recall without input adapters. The AUC values can be found in Table II.

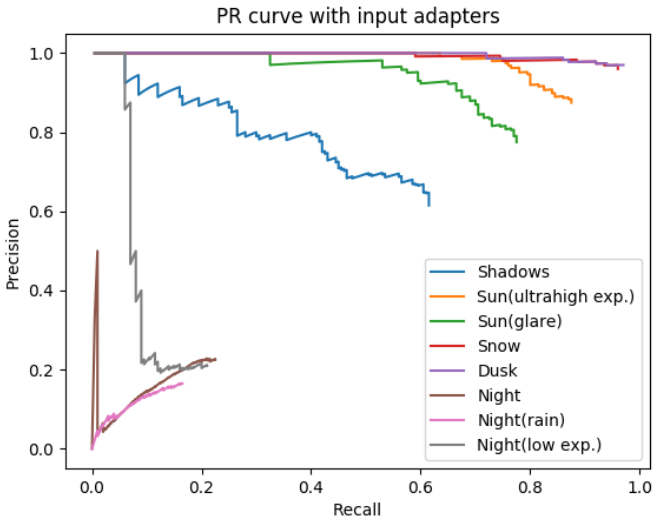


Fig. 8. RobotCar topological localisation Precision-Recall with input adapters. The AUC values can be found in Table II.

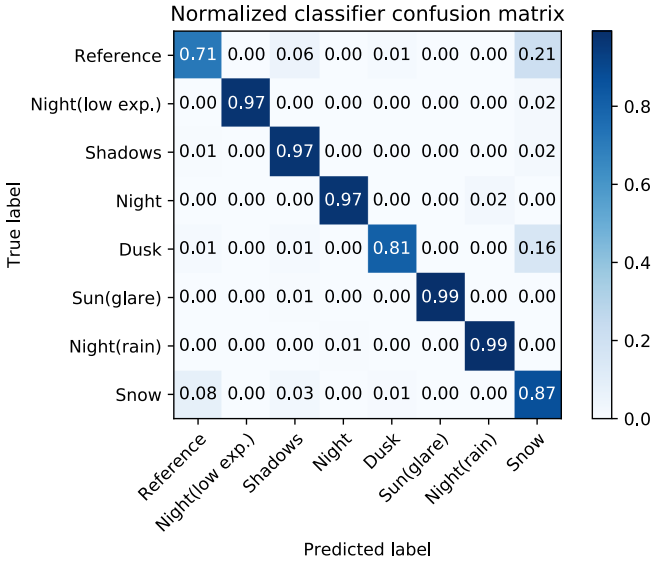


Fig. 9. RobotCar Condition Classifier confusion matrix. Our condition classifier achieves a 91% overall accuracy rate.

TABLE I

BDD SEGMENTATION MEAN INTERSECTION OVER UNION (MIOU)

No adapter	Deeplab-all	Indiv. Deeplabs	Indiv. adapters(**ours**)
0.4070	0.4100	0.4319	**0.4503**

VI. CONCLUSIONS

To prevent performance of computer vision tasks from degrading quickly and often catastrophically when input conditions change, we have presented a domain adaptation system that uses light-weight input adapters to pre-processes input images, irrespective of their appearance, in a way that makes them compatible with off-the-shelf computer vision tasks that are trained only on inputs with ideal conditions. No fine-tuning is performed on the off-the-shelf models, and the system is capable of incrementally training new input adapters in a self-supervised fashion, using the computer vision tasks as supervisors, when the input domain differs significantly from previously seen domains. We report large improvements in semantic segmentation and topological localization performance on two popular datasets, RobotCar and BDD. This work is presented as a framework, and each end-to-end differentiable component may be replaced with a better-performing counterpart, or with one that is better-suited for the task at hand, if available. Additionally, the training process may be extended to work with an arbitrary number of supervisory signals. Finally, our on-line training regimen benefits from convergence times that decrease as a function of the number of domains trained on.

VII. ACKNOWLEDGEMENTS

This work was supported by a Oxford-Google DeepMind Graduate Scholarship and EPSRC/UK Research and Innovation Programme Grant EP/M019918/1.

TABLE II

ROBOTCAR TOPOLOGICAL LOCALISATION AREA UNDER CURVE (AUC)

Method	Shadows	Sun(glare)	Snow	Dusk	Night	Night(rain)	Night(low exp.)	Sun(ultrahigh exp.)	Mean
No adapters	0.2928	0.3949	0.9263	**0.9710**	0.0009	0.0000	0.0719	0.7043	0.4202
With adapters(**ours**)	**0.4965**	**0.7404**	**0.9494**	0.9605	**0.0374**	**0.0191**	**0.0981**	**0.8593**	**0.5200**

TABLE III

ROBOTCAR SEGMENTATION MEAN INTERSECTION OVER UNION (MIOU)

Method	Shadows	Sun(glare)	Snow	Dusk	Night	Night(rain)	Night(low exp.)	*Sun(ultrahigh exp.)*	Mean
Baseline	0.6014	0.4316	0.5677	0.6156	0.1404	0.0859	0.1850	*0.4423*	0.3837
Deeplab-all	0.5375	0.4712	0.5055	0.5372	0.3465	0.3572	0.3593	*0.4821*	0.4495
Indiv. deeplabs	0.5594	0.4948	0.5303	0.5656	0.3948	0.4138	0.4184	*0.5120*	0.4861
Indiv. adapters(**ours**)	**0.6292**	**0.6419**	**0.6525**	**0.6327**	**0.5301**	**0.5627**	**0.5136**	***0.5500***	**0.5891**

REFERENCES

[1] D. Dai and L. V. Gool, "Dark model adaptation: Semantic image segmentation from daytime to nighttime," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, Nov 2018, pp. 3819–3824.

[2] H. Porav, W. Maddern, and P. Newman, "Adversarial training for adverse conditions: Robust metric localisation using appearance transfer," *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1011–1018, 2018.

[3] A. Anoosheh, T. Sattler, R. Timofte, M. Pollefeys, and L. V. Gool, "Night-to-day image translation for retrieval-based localization," *CoRR*, vol. abs/1809.09767, 2018.

[4] L. Ma, X. Jia, S. Georgoulis, T. Tuytelaars, and L. V. Gool, "Exemplar guided unsupervised image-to-image translation with semantic consistency," in *International Conference on Learning Representations*, 2019.

[5] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. Efros, and T. Darrell, "CyCADA: Cycle-consistent adversarial domain adaptation," in *Proceedings of the 35th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, J. Dy and A. Krause, Eds., vol. 80. Stockholmsmssan, Stockholm Sweden: PMLR, 10–15 Jul 2018, pp. 1989–1998.

[6] M. Limmer and H. P. A. Lensch, "Infrared colorization using deep convolutional neural networks," *CoRR*, vol. abs/1604.02245, 2016.

[7] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, "Deep domain confusion: Maximizing for domain invariance," *CoRR*, vol. abs/1412.3474, 2014.

[8] M. Wulfmeier, A. Bewley, and I. Posner, "Addressing appearance change in outdoor robotics with adversarial domain adaptation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, October 2017.

[9] Z. Ren and Y. J. Lee, "Cross-domain self-supervised multi-task feature learning using synthetic imagery," *CoRR*, vol. abs/1711.09082, 2017.

[10] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, "Adaptive mixtures of local experts," *Neural Comput.*, vol. 3, no. 1, pp. 79–87, Mar. 1991.

[11] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proceedings of the 1st Annual Conference on Robot Learning*, 2017, pp. 1–16.

[12] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[13] S. Sankar, Y. Balaji, A. Jain, S.-N. Lim, and R. Chellappa, "Learning from synthetic data: Addressing domain shift for semantic segmentation," 06 2018, pp. 3752–3761.

[14] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *ECCV*, 2018.

[15] J. Fu, Y. Wang, and H. Lu, "Stacked deconvolutional network for semantic segmentation," *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, 2017.

[16] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[17] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3213–3223.

[18] M. Cummins and P. Newman, "FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance," *The International Journal of Robotics Research*, vol. 27, no. 6, pp. 647–665, 2008.

[19] M. J. Milford and G. F. Wyeth, "Seqslam: Visual route-based navigation for sunny summer days and stormy winter nights," in *2012 IEEE International Conference on Robotics and Automation*, May 2012, pp. 1643–1649.

[20] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[21] Y.-H. Tsai, W.-C. Hung, S. Schulter, K. Sohn, M.-H. Yang, and M. K. Chandraker, "Learning to adapt structured output space for semantic segmentation," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7472–7481, 2018.

[22] N. Souly, C. Spampinato, and M. Shah, "Semi and weakly supervised semantic segmentation using generative adversarial network," *CoRR*, vol. abs/1703.09695, 2017.

[23] Z. Murez, S. Kolouri, D. J. Kriegman, R. Ramamoorthi, and K. Kim, "Image to image translation for domain adaptation," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4500–4509, 2018.

[24] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017.

[25] A. Dundar, M.-Y. Liu, T.-C. Wang, J. Zedlewski, and J. Kautz, "Domain stylization: A strong, simple baseline for synthetic to real image domain adaptation," *arXiv preprint arXiv:1807.09384*, 2018.

[26] C. Sakaridis, D. Dai, and L. Van Gool, "Semantic nighttime image segmentation with synthetic stylized data, gradual adaptation and uncertainty-aware evaluation," *ArXiv e-prints*, 2019.

[27] L. Clement and J. Kelly, "How to train a cat: Learning canonical appearance transformations for direct visual localization under illumination change," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2447–2454, July 2018.

[28] M. Wulfmeier, A. Bewley, and I. Posner, "Incremental adversarial domain adaptation for continually changing environments," in *International Conference on Robotics and Automation (ICRA)*, 2018.

[29] M. Brki, M. Dymczyk, I. Gilitschenski, C. Cadena, R. Siegwart, and J. Nieto, "Map management for efficient long-term visual localization in outdoor environments," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, June 2018, pp. 682–688.

[30] R. Gong, W. Li, Y. Chen, and L. V. Gool, "Dlow: Domain flow for adaptation and generalization," *CoRR*, vol. abs/1812.05418, 2018.

[31] H. Germain, G. Bourmaud, and V. Lepetit, "Efficient condition-based representations for long-term visual localization," *CoRR*, vol. abs/1812.03707, 2018.

[32] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 Year, 1000km: The Oxford RobotCar Dataset," *The International Journal of Robotics Research (IJRR)*, vol. 36, no. 1, pp. 3–15, 2017.

[33] ——, "1 Year, 1000km: The Oxford RobotCar Dataset," *The International Journal of Robotics Research (IJRR)*, vol. 36, no. 1, pp. 3–15, 2017.

[34] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds. Cham: Springer International Publishing, 2015, pp. 234–241.

[35] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[36] R. Arandjelovi, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 6, pp. 1437–1451, June 2018.

[37] F. Yu, W. Xian, Y. Chen, F. Liu, M. Liao, V. Madhavan, and T. Darrell, "BDD100K: A diverse driving video database with scalable annotation tooling," *CoRR*, vol. abs/1805.04687, 2018.

3.2 Summary of the Results

The presented publication makes two contributions in the context of this thesis.

Firstly, we prevent a significant decrease in segmentation performance under adverse weather and lighting conditions. This was achieved by implementing a bank of condition-dependent input adapters to map the input images to an appearance-invariant representation optimal for the pretrained segmentation network. The intuitive reasoning behind this idea is that the ground-truth semantic segmentation is invariant to appearance changes, and thus there should exist an underlying shared representation (i.e. the optimisation objective) that is independent of the input condition. Concretely, we have tested four differently-trained models against data of adverse conditions:

- A baseline model that was trained on a reference (i.e. overcast) dataset.
- A model that was trained on a combined dataset containing all conditions.
- Multiple condition-dependent models which were each trained on a dataset containing only one respective condition. A domain classifier selects the correct model according to the input condition.
- The presented framework, which includes the domain classifier to select an adapter for the respective input condition and performs semantic segmentation on the appearance-invariant representation using the baseline model.

Several observations are made from the results in the publication:

- The segmentation performance is negatively affected when the input condition does not align with the training condition, as demonstrated by the baseline model.
- The individually-trained models outperform the combined model, which is expected since the DNNs can devote their entire capacity to a single condition.

- The approach presented in the publication outperforms all other cases because we explicitly direct the network to learn a useful mid-level representation.

Secondly, the framework described above is trained in a self-supervised fashion. We again leverage the fact that the ground-truth semantic segmentation is invariant to the input condition and employ a cycle-consistency framework to change the appearance of the image. It is then possible to pair, for instance, a generated nighttime image with the ground-truth semantic label of the overcast image. This significantly reduces the labelling effort for generating training data for multiple conditions since the typical approach creates labels for every condition manually [55], [77].

3.3 Conclusion

This chapter extended the general case of semantic segmentation under overcast conditions towards a wide range of environmental conditions. More specifically, we have resolved scalability and robustness issues by leveraging the fact that the ground-truth semantic segmentation is invariant to the input conditions.

However, as mentioned previously in Chapter 1, the output representation is limited in terms of its direct usefulness for decision making and planning. Firstly, the pixel-wise semantic segmentation does not naturally allow for high-level reasoning concerning driving actions. We resolve this issue by presenting a graph-based framework, which incorporates segmented entities and can be linked directly to the decision-making process in Chapter 4. Secondly, the resulting segmentation does not include the semantics of the road markings in the scene, which are crucial for guiding traffic participants. In Chapter 5 and 6, we obtain these cues under different conditions in a self-supervised manner. Lastly, the perspective of the front-facing camera image is inconvenient for many tasks of the autonomous driving pipeline, including planning. Therefore, we learn an accurate bird's-eye-view mapping and show its usefulness with regard to high-level scene understanding in Chapter 7.

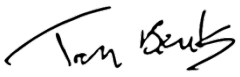
Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor **(only required where there isn't already a statement of contribution within the paper itself).**

Title of Paper	Don't Worry About the Weather: Unsupervised Condition-Dependent Domain Adaptation
Publication Status	Published
Publication Details	H. Porav, **T. Bruls**, and P. Newman, "Don't worry about the weather: Unsupervised condition-dependent domain adaptation", in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Oct. 2019, pp. 33-40.

Student Confirmation

Student Name:	Tom Adriaan Hubert Bruls		
Contribution to the Paper	Contributions included: - Generating the initial ideas regarding a canonical representation and multi-conditional dataset generation. - Preparing and processing the semantic segmentation data. - General editing of the paper. The overall paper emerged as a product of discussions and collaboration with my co-authors.		
Signature		Date	10-05-2020

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Professor Paul Newman			
Supervisor comments			
Signature		Date	11-05-2020

This completed form should be included in the thesis, at the end of the relevant chapter.

4

Representations for Integration: Scene Graphs

Contents

4.1 Hierarchical Road Scene Understanding 40
4.2 Prerequisites for Scene Graphs 42
4.2.1 Representations for Conduct 42
4.2.2 Representations for Overview 45

As discussed in Chapter 3, deep semantic segmentation networks offer versatility in describing a wide variety of scenes, but the pixel-wise output representation is limited in its direct deployment for navigating through complex urban environments. This and subsequent chapters present more effective representations for road scene understanding, which follow the principles outlined in Section 2.1.1, to improve integration, understanding of conduct, and overview.

In order to facilitate explainable and interpretable decision making in autonomous driving pipelines, a hierarchical, graph-based representation, called the *scene graph*, is demonstrated in Section 4.1. This representation is ideally suited for describing highly-structured urban environments. The graphs integrate low-level perception cues, such as segmented road markings obtained by DNNs, and domain knowledge regarding road construction to describe higher-level concepts such as the road

layout. This enables the framework to offer a balance between an appropriate abstraction level for decision making and the versatility to represent a multitude of scenes from different viewpoints.

Several prerequisites are required for generating scene graphs such as (semantic understanding of) the road markings, accurate IPM, and curb detection [78]. We refer to the first two as *representations for conduct* and *representations for overview*, respectively. Obtaining these introduces various challenges, which we discuss in Section 4.2. We aim to overcome these challenges in the subsequent chapters.

In summary, this chapter makes the following principal contribution:

- An integration and discussion of the obtained binary road markings into a higher-level scene understanding framework for inferring the road layout (T-1b).

Additionally, this chapter makes the following supporting contribution in collaboration:

- A demonstration of a hybrid framework for high-level road scene understanding, which combines object-centric perception, pixel-wise learning, and prior domain knowledge into a graph-based description of the road (T-1a).

4.1 Hierarchical Road Scene Understanding

This section presents our publication on a hierarchical understanding of urban road layouts, which is reproduced in Appendix A.

The presented approach models the road layout of a scene using a hierarchical graph containing semantic and spatial constraints, similar to [79]. Road layouts are inherently constructed hierarchically and can therefore be described in a bottom-up fashion starting from low-level cues such as lane separators and building up towards lanes, roads, and ultimately the entire scene.

Figure 4.1 shows an example scene graph. The scene graphs are constructed from pixel-wise segmentations in the front-facing image. In particular, we include binary road markings (Figure 4.1(a), obtained in Chapter 5) and curbs [78], but the

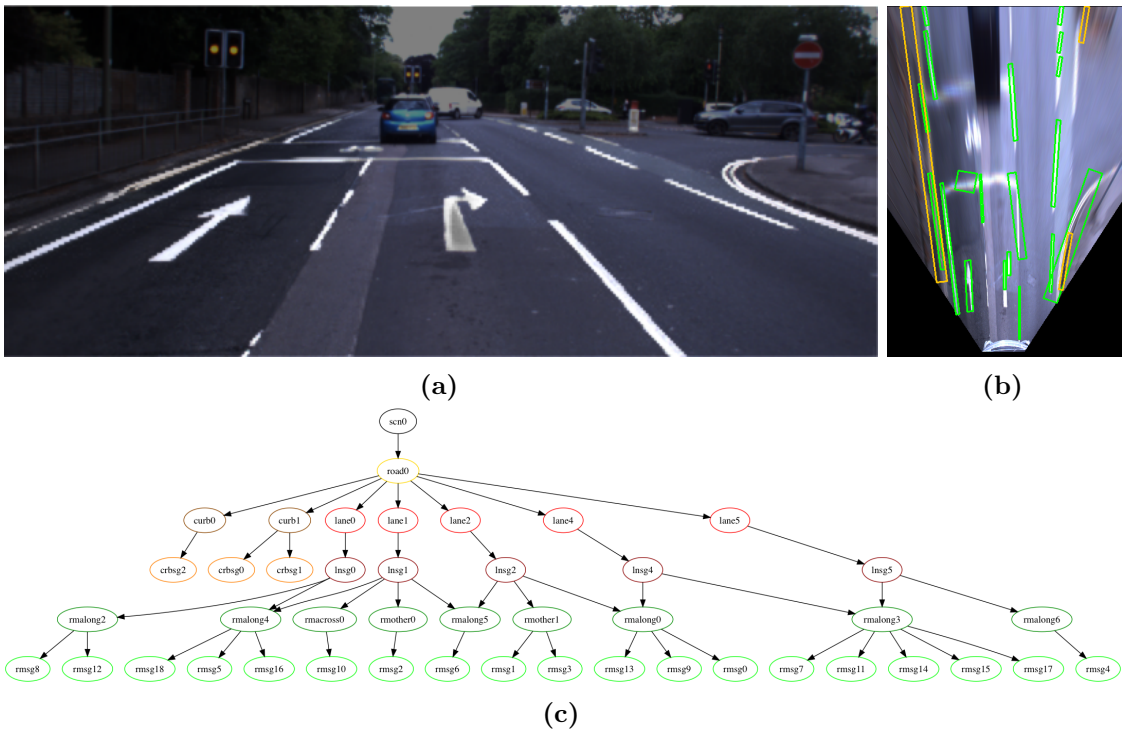


Figure 4.1: An example of a generated scene graph. The road markings are segmented in the front-facing image with a DNN, **(a)**, and transformed into a bird's-eye view, **(b)**. A hierarchical road layout description, **(c)**, is constructed from the partial segmentations (road markings in *green*, curbs in *yellow*), a probabilistic grammar, and a spatial relational model. More explanations and details are provided in the reproduced publication in Appendix A.

framework can be extended to include additional low-level cues such as traffic signs and traffic participants. These segmentations, as well as their front-facing images, are transformed into a bird's-eye view (Figure 4.1(b)), which makes reasoning more convenient as this is closer to the vehicle's action space than the front-facing perspective [80], [81]. The road layout is reconstructed from these entities in this view using a learned probabilistic context-free grammar and a learned spatial relational model. The remaining chapters of this thesis focus on obtaining the aforementioned prerequisites: (a semantic understanding of) the road markings and an accurate bird's-eye view.

The resulting graph-based representation can be employed for cost-based planning, inferring object classes, or reasoning about missing and occluded parts by leveraging domain knowledge (e.g. the UK Highway Code). More importantly,

it allows for explaining the vehicle's behaviour and decision making, which is paramount for real-world deployment and adoption [82].

4.2 Prerequisites for Scene Graphs

In this section, we discuss the prerequisites required for building accurate scene graphs. More specifically, we introduce *representations for conduct* (i.e. road markings) and *representations for overview* (i.e. IPM). We then explain the challenges that arise when obtaining these representations.

4.2.1 Representations for Conduct

We provide a definition of road markings before discussing the challenges that arise when performing road marking segmentation and classification.

Road Markings

We define a road marking as a semantic instance that imposes a particular driving behaviour. These instances are generally formed by a collection of (linear) road marking segments configured according to specified distances and angles. For example, multiple adjacent linear segments connected at a defined, constant angle form a zig-zag marking, as illustrated in Figure 4.2(c), to warn drivers of an upcoming pedestrian crossing.

Other than camera-based methods, LiDAR reflectance is a popular sensor modality to achieve a semantic understanding of the road markings since they are highly reflective by design. Such approaches [83], [84] might perform better than camera-based methods under certain circumstances since they are not affected by the image perspective and illumination. However, LiDAR is relatively expensive, and these methods generally do not function online (i.e. aggregate point clouds over time). They are, therefore, better suited for mapping purposes.

Road markings in images can be represented in various ways, as visualised in Figure 4.2. We use the following definitions:

(a) **(b)** **(c)**

Figure 4.2: Various ways of representing road markings. Road marking detection draws bounding boxes around letters and symbols with an indication of the type [85], **(a)**. Road marking segmentation performs a pixel-wise binary segmentation of the road markings in the scene, **(b)**. Road marking classification retrieves instances of road markings with a particular semantic meaning, **(c)**.

- Road marking detection is the task of detecting a certain road marking type, which is generally enclosed by a bounding box [86], [87], similar to object detection. This works well for letters and symbols but is inconvenient for lane structures (e.g. double boundaries).

- Road marking segmentation is the task of segmenting the road markings in an image in a pixel-wise fashion. In contrast to detection, this works well for all types of road markings. However, the resulting pixel-wise representation does not directly support high-level reasoning. In this thesis, road marking segmentation refers to the binary case studied in Chapter 5.

- Road marking classification is the task of classifying semantic instances of road markings that impose particular road rules. The ideal output of such a task is road marking instances, as in Section 6.1, which can be integrated directly into scene understanding frameworks. However, we also refer to road marking classification for the pixel-wise output of Section 6.2 as the pixels are classified according to their semantic meaning.

Although the pixel-wise representations obtained by DNNs require additional post-processing steps to support reasoning about complex driving actions, we choose this as a starting point because of its versatility. State-of-the-art methods [88] incorporate higher-level reasoning directly in the DNN while simultaneously minimising the DNN size.

(a) **(b)**

Figure 4.3: The road markings are classified semantically and displayed in different colours, **(b)**, for a complex urban intersection, **(a)**. We ideally retrieve this mapping directly from the image and include the result in the scene graph to guide planning and decision making.

Targeted literature studies are provided in the reproduced publications in Chapter 5 and 6.

Challenges

The scene graphs in the publication are constructed from binary road marking segmentations, but these provide limited information in real-world scenarios. Although they allow for understanding the lane structures, they do not capture the underlying semantic meaning. These semantics are crucial as they dictate the road rules for safe driving behaviour, and thus we strive to extend the current framework by incorporating them.

We would ideally distinguish and retrieve the semantic classes of the road markings directly from the image, as shown in Figure 4.3. However, this is complicated because the appearance of the road markings in the image is affected by the image perspective, occlusions, and weather and lighting conditions. Consequently, heuristic approaches for road marking classification do not perform satisfactorily.

DNNs recently resolved some of these limitations and set a new state-of-the-art at the cost of requiring vast quantities of expensive pixel-wise labelled data. We circumvent this limitation throughout several chapters of this thesis by introducing self-supervised frameworks. We would ideally generate training pairs automatically for learning the mapping shown in Figure 4.3 directly, but this only became possible recently through progress in computer vision. Such an approach is presented in Chapter 6. Therefore, a binary mapping was learned first in a self-supervised way

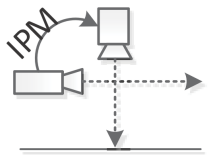


Figure 4.4: The IPM transformation adjusts the camera position from the front-facing perspective towards a top-down perspective [89].

in Chapter 5 by leveraging additional sensor modalities and domain knowledge. This mapping was later extended to full road marking classification in Chapter 6 by incorporating additional domain knowledge regarding road marking construction.

4.2.2 Representations for Overview

We conceptually introduce IPM before discussing the limitations it introduces for generating accurate scene graphs.

Inverse Perspective Mapping (IPM)

The perspective of the front-facing camera distorts distances and depth significantly. In order to address this, many tasks in the autonomous driving pipeline are commonly performed in a bird's-eye view. This view is more closely related to the vehicle's action space and provides convenient integration of various sensor modalities such as images and LiDAR point clouds.

A homography matrix, which defines the relationship between the pixels in the front-facing camera and the bird's-eye view, is traditionally computed by assuming that the road surface is planar [90]. This matrix can either be estimated from point correspondences in the images or from the camera and vehicle calibration. Conceptually, the transformation moves the camera position from the front-facing perspective towards a top-down perspective, as shown in Figure 4.4.

A more extensive literature study of various improvements for IPM is provided in the publication in Chapter 7.

(a) **(b)**

Figure 4.5: Scenarios in which the quality of the generated scene graph is negatively affected. Road markings cannot be detected in overexposed areas of the road surface, consequently we cannot reason about these parts of the scene. For example, road markings can become difficult to distinguish, leading to the failed detection of the right bicycle lane, **(a)**. Furthermore, the shape and form of objects farther away such as the zig-zag markings become distorted by the IPM transformation, and thus they might not be segmented properly, **(b)**. The blue, opaque boxes indicate missing parts of the scene graphs due to these issues. We resolve these scenarios in the publication in Section 7.1.

Challenges

Several factors, of which we will discuss two here in more detail, potentially limit the quality of the generated scene graph in this view.

Firstly, since the scene graph is reconstructed from segmented entities in images only, occluded regions of the road surface (either by objects or weather and lighting conditions) can lead to missing information or misinterpretation. An example of this is given in Figure 4.5(a), where the road markings are undetectable in overexposed regions of the image. Secondly, IPM only works well in practice in the immediate

proximity of the vehicle (assuming the road surface is planar) as objects in the distance are deformed unnaturally by this mapping [91]. This is demonstrated in Figure 4.5(b), where the shape of the zig-zag markings is deformed to such an extent that they have become (1) hard to segment and (2) do not accurately represent the real world. Both cases limit the distance at which we can reliably generate the scene graph. This gives an autonomous vehicle less time to alter its behaviour accordingly, which is critical for safety.

Small inaccuracies in this mapping can thus lead to significant qualitative differences in the semantic interpretation of scenes. These qualitative differences can manifest themselves in many ways, including missing lanes and late detection of stop lines (or other critical road markings). We present a learned IPM, called *boosted IPM*, to overcome these limitations in Chapter 7.

5

Representations for Conduct I: Road Marking Segmentation

Contents

5.1 Publication 54
5.2 Approximated Road Marking Labels 63
5.2.1 Further Details 63
5.2.2 Further Results 65
5.2.3 Further Discussion 66
5.3 DNN Road Marking Segmentation 67
5.3.1 Further Details 67
5.3.2 Further Results 68
5.3.3 Further Discussion 72
5.4 DNN Road Marking Segmentation under Rainy Conditions 73
5.4.1 De-Raining Images 73
5.4.2 Summary of the Results 74
5.5 Conclusion 76

An accurate understanding of the road markings in a traffic scene is critical for autonomous vehicle operation. Their underlying meaning provides rules and guidance to all traffic participants and warns them of potentially dangerous situations. Hence, the road markings, together with other relevant objects such as curbs and traffic signs, serve as low-level cues in frameworks that aim to achieve high-level scene understanding in a bottom-up fashion, as demonstrated in Chapter

(a) **(b)** **(c)**

Figure 5.1: Road marking segmentation and an approximated training label for an urban road scene. We define road marking segmentation as a binary segmentation of the road markings, **(b)**, in the scene, **(a)**. Our aim is to generate an *approximated road marking label*, **(c)**, which is not necessarily equal to the ground truth but sufficient for training DNNs in a self-supervised way. The visualized label fails to include the separator on the right, for example.

4. However, the pixel-wise representation obtained in Chapter 3 lacks the level of detail required for a semantic understanding of these crucial elements. In the next chapter, we present an efficient and scalable framework for state-of-the-art road marking classification to achieve this kind of reasoning and understanding at large scales, building upon the binary segmentation presented in this chapter.

As mentioned previously, we restrict ourselves during deployment to the use of front-facing camera images due to the low cost yet information richness of this sensor modality and the availability of well-established image processing techniques. We define the problem of road marking segmentation as a binary segmentation of the pixels in an input image that belong to road markings painted on the road surface (Figure 5.1b). We specifically solve the binary segmentation problem first because this opens up the possibility of self-supervised learning. Classification is then achieved by a subsequent model-driven optimization step, which leverages additional domain knowledge regarding road marking construction, in Section 6.1. In Section 6.2, we present an alternative data-driven approach that directly classifies the road markings in the image semantically by using image-to-image translation techniques that have only been published recently (i.e. after the publication of the work presented in this chapter).

Road marking segmentation is a challenging problem for several reasons. Firstly, the aim is to detect the entire collection of painted markings on the road surface, not just the separators which mark the different lanes. These come in various shapes and types (e.g. arrows, letters, and symbols), can be country-specific, and degrade over

Figure 5.2: Road marking segmentation is challenging for several reasons such as occlusions, (**a**), degradation, (**b**), overexposure, (**c**), or combinations of these, (**d**). Heuristic methods most likely fail under these circumstances. DNNs might resolve these limitations when adequate training data is available and by using the global scene context.

time. Secondly, visual limitations such as occlusions, varying lighting, and changing weather conditions are common. Some examples of challenging scenarios are given in Figure 5.2. Early approaches have proposed hand-crafted features for road marking segmentation [92]. However, these often lead to unsatisfactory performance during real-world deployment due to their heuristic nature in combination with the aforementioned challenges.

In order to solve some of these limitations, road marking segmentation has recently been posed as a semantic segmentation problem [15], [88], in which state-of-the-art DNNs are trained to provide pixel-wise outputs. Their advantage is twofold. Firstly, they are specifically designed to leverage the global scene context to improve the segmentation. For instance, a row of parked cars is usually enclosed by road markings indicating the parking areas. Secondly, they are robust to spatial deformations, degradation, and partial occlusion when adequate training data is available.

Nevertheless, vast quantities of training samples are required to achieve high performance and proper generalization. Most urban driving datasets such as the

KITTI [50] dataset, Cityscapes [53], and the Oxford RobotCar dataset [61] do not provide ground-truth labels for small classes such as road markings. The reason for this is that manually labelling these classes is extremely labour intensive due to the required pixel-level detail and the challenges of dealing with the aforementioned visual issues. The first large-scale dataset containing pixel-wise semantic labels for road markings [14] was only released after the publication of our work. Despite this, it remains practically infeasible to expand such datasets manually to encompass all of the environments and conditions that could be encountered by vehicles in the real world.

In order to reduce the labelling effort, we have published a method for generating approximated road marking labels, which are referred to as *annotations* in the reproduced publication (Section 5.1), automatically by leveraging complementary sensors and domain knowledge. These approximated labels are not necessarily equivalent to the ground-truth labels, as illustrated in Figure 5.1, but are sufficient for training a DNN in a self-supervised way. We argue that pixel-valued intensity-based road marking segmentation approaches are not sufficient for this purpose as they are likely to fail (e.g in the case of overexposure). We, therefore, use LiDAR as a complementary modality, which has two important benefits: (1) LiDAR is an active sensor and therefore not affected by external illumination, and (2) LiDAR measures reflectance and is thus able to exploit the fact that road markings are highly reflective [83], [84], [93]. We expand upon the material presented in the publication with regard to the generation of the approximated labels in Section 5.2.

The publication further demonstrates that the approximated labels are useful in multiple ways when training DNNs for road marking segmentation. Firstly, the segmentation quality can exceed the quality of the label towards segmenting the full ground truth if proper regularization techniques are employed. Secondly, the approximated labels can boost performance in other domains where only a limited number of ground-truth labels (and no LiDAR data) are available, either through pretraining or by adding them directly to the training dataset. Crucially, the trained networks only require input from a monocular camera during deployment and run

online. We expand upon the material presented in the publication with regard to the road marking segmentation in Section 5.3.

Although the publication in Section 5.1 demonstrates the ability of the system to work under various environmental conditions, it does not consider contamination of the camera lens (e.g. raindrops). This significantly distorts the view and consequently deteriorates the road marking segmentation performance more drastically. A solution is demonstrated in Section 5.4 (and Appendix B) which preprocesses the image (i.e. de-rains it) so that the input to the segmentation network is almost equivalent to an overcast image [22]. This restores the segmentation performance.

In summary, this chapter makes the following principal contributions:

- A method for generating approximated road marking labels which are sufficient for training DNNs in a self-supervised way by leveraging complementary sensor modalities and domain knowledge, thereby avoiding expensive manual labelling (T-2a and T-2b).

- A framework for online road marking segmentation in complex urban environments using a monocular camera that does not rely on preprocessing steps, predefined models, or manually labelled data (T-1b).

- An evaluation and ablation study of the domain knowledge captured by the approximated road marking labels when transferring it to domains where only a few manual labels are available (T-1b and T-2a).

- An evaluation of road marking segmentation in rainy conditions with lens distortions (T-1b).

Additionally, this chapter makes the following supporting contribution in collaboration:

- A demonstration of an image de-raining approach to counteract lens distortions and thereby restore the road marking segmentation (T-3b).

5.1 Publication

This section contains a reproduction of the following publication:

> [18] **T. Bruls**, W. Maddern, A. A. Morye, and P. Newman, "Mark yourself: Road marking segmentation via weakly-supervised annotations from multimodal data", in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 1863–1870[1].

[1] video accompanying the publication with extensive explanations and results: `https://www.youtube.com/watch?v=2vR00jDYgTI`

Mark Yourself: Road Marking Segmentation via Weakly-Supervised Annotations from Multimodal Data

Tom Bruls, Will Maddern, Akshay A. Morye, and Paul Newman

***Abstract*— This paper presents a weakly-supervised learning system for real-time road marking detection using images of complex urban environments obtained from a monocular camera. We avoid expensive manual labelling by exploiting additional sensor modalities to generate large quantities of annotated images in a weakly-supervised way, which are then used to train a deep semantic segmentation network. At run time, the road markings in the scene are detected in real time in a variety of traffic situations and under different lighting and weather conditions without relying on any preprocessing steps or predefined models. We achieve reliable qualitative performance on the Oxford RobotCar dataset, and demonstrate quantitatively on the CamVid dataset that exploiting these annotations significantly reduces the required labelling effort and improves performance.**

I. Introduction

Autonomous vehicles need to understand their workspace for informed decision making and safe navigation in complex urban settings. In contrast to recently developed end-to-end approaches for autonomous driving [1], mediated approaches detect important objects in the scene separately to build a combined, real-time model of the environment that can be employed for navigation and operational purposes. In urban environments, the collection of all painted road markings (e.g. Fig. 1) is critical in such models: their underlying meaning provides rules and guidance to all traffic participants and warns them of potentially dangerous situations. This paper presents a first step towards interpretation of these road rules by presenting a framework for road marking detection in a variety of traffic, lighting, and weather conditions.

In the domain of autonomous vehicles, highly detailed mapping services such as Google Maps, HERE Maps, OpenStreetMap, etc., include road graphs that can support scene understanding. However, relying solely on these can cause problems whenever the traffic situation is updated, or when unmapped places are visited. Even in a future of connected cars, real-time detection and interpretation of road markings will remain an important cue for high-level scene understanding and thereby aid planning, localization [2], and mapping [3].

In this paper, we detect not only separators that mark the different lanes, but the collection of all painted markings on the road surface that dictate the traffic rules for that particular urban setting. Detecting and interpreting these is a more complex problem than lane detection. In general, proposed solutions in that area do not extend easily to the detection of a bigger variety of road markings.

Authors are from the Oxford Robotics Institute, Dept. Engineering Science, University of Oxford, UK. {tombruls, pnewman}@robots.ox.ac.uk

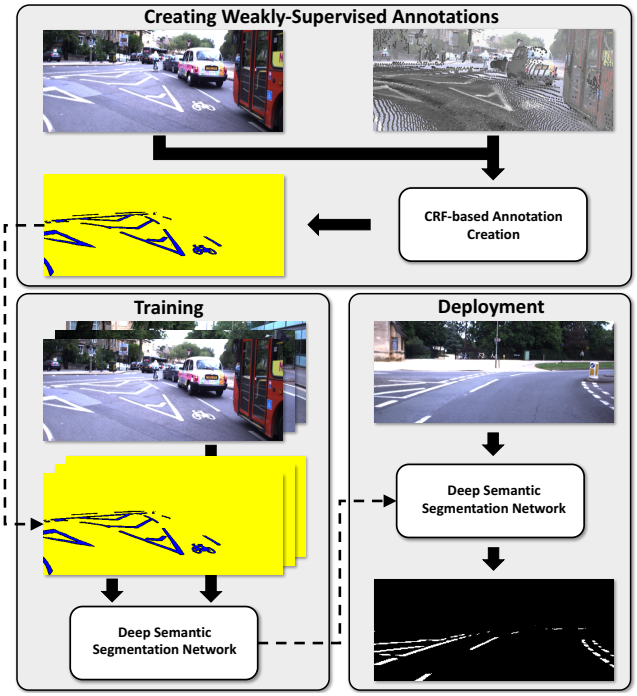


Fig. 1. Road marking detection using weakly-supervised annotations. A LiDAR point cloud of reflectance values is combined with a monocular image to generate road marking annotations in a weakly-supervised way using a conditional random field approach (Section III). A deep semantic segmentation network is then trained using these annotations and the corresponding images (Section IV). During deployment the network performs road marking detection in real time without any additional processing steps using only a monocular camera (Section V).

Road marking detection is a challenging problem for several reasons. Firstly, a proposed method has to cope with occlusions, varying lighting, and changing weather conditions. Secondly, road markings are often degraded and vary in sorts and shapes between countries. Lastly, there are no large datasets available that contain accurate ground-truth labels for road markings. Most datasets for urban scenarios such as KITTI [4], Cityscapes [5], and the Oxford RobotCar dataset [6] do not provide the level of detail that is required for segmenting such small classes.

Road marking detection in images can be posed as a semantic segmentation problem. State-of-the-art methods for these tasks implement deep networks, which are able to learn specific scene context and thereby cope with the challenges stated above, as long as sufficient training data

is available. Although some networks have been trained for road marking recognition [7], [8], their applicability remains limited because of the current lack of ground-truth labels.

Manual generation of these ground-truth labels for semantic segmentation tasks is extremely labour expensive, because of the required pixel-level detail in combination with the aforementioned visual issues. Therefore, we present a method for creating annotations in a weakly-supervised way, by leveraging complementary sensors mounted on the vehicle. We utilize these annotations to train a deep semantic segmentation network (inspired by U-Net [9]) for road marking detection using only a monocular camera. The annotations do not necessarily capture all the road markings in the image perfectly, but are sufficient for training purposes as explained in Section III-C.

We present qualitative results of our approach in a variety of traffic, lighting, and weather conditions on the RobotCar dataset. Furthermore, we show quantitatively that exploiting the weakly-supervised RobotCar annotations significantly reduces the required labelling effort and improves detection performance on the CamVid dataset [10].

We make the following contributions in this paper:

- We present a method for creating road marking annotations in a weakly-supervised way by using complementary sensor modalities. These are used for training a deep semantic segmentation network, thereby avoiding expensive manual labelling.
- We introduce a real-time framework for road marking detection in complex urban settings using a monocular camera without relying on any preprocessing steps or predefined models. This method performs reliably in a wide variety of traffic, lighting, and weather conditions.

The combination of these contributions (see Fig. 1) provides a first step towards road marking classification in datasets without ground-truth labels to support high-level scene understanding, mapping, and planning.

II. Related Work

Our work on road marking segmentation based on weakly-supervised learning from multimodal data is mainly related to work in the area of road marking detection — which we discuss first. We further discuss related work in the areas of lane detection, semantic segmentation, and automatic label generation.

1) Road Marking Detection: Work on road marking detection can generally be distinguished by the used sensor modalities (e.g. camera or LiDAR) and whether learning algorithms are applied (unsupervised or supervised).

Unsupervised camera-based road marking detection systems often follow a four stage pipeline: preprocessing, filtering/binarization, feature extraction, and (rule-based) classification. An early evaluation of several techniques is given in [11]. While effective in moderate environments, these approaches fail in the presence of extreme lighting conditions and shadows. Other disadvantages include hand-crafted features used for template matching [12] and shape-based classification, which both perform badly in the presence of occlusions.

Supervised approaches often use a similar pipeline with the exception that the last step is replaced by a supervised classification algorithm. Popular classifiers include random forests [13], SVMs [14], shallow neural nets [15], and OCR for text recognition [16]. Computed features include HOG and Hu spatial moments, which are rotation and scale invariant and thus perform better under challenging conditions and occlusions. Other approaches [17], [18], do not classify detected road markings independently, but take the spatial configuration of the entire scene into account. This is preferable because road markings are often found in the same spatial configuration.

More recently, deep networks have been successfully introduced for road marking recognition [7], [19] or purely for classification [8]. However, these approaches either implement additional preprocessing algorithms or require detected road markings as an input, because of the current lack of ground-truth road marking labels in large-scale urban datasets. We resolve this issue by creating annotations in a weakly-supervised way.

Lately, the use of LiDAR reflectance values has become more popular as an indication for road markings, since they are not affected by varying lighting. Most solutions generate an interpolated 2D reflectance image [20], so that well-known image processing techniques can be applied. In contrast, the latest approaches work directly on the point cloud [21]. However, because LiDARs are still relatively expensive, these approaches are mainly applied for mapping purposes and not for real-time road marking detection. Therefore, we make use of LiDAR sensors only during the offline annotation creation, and rely solely on a monocular camera during deployment.

2) Lane Detection: Most lane detection systems consist of detection, model fitting, and tracking stages, as summarized in [22]. More recently, deep networks [23], [24] have been proposed, because they perform better under challenging conditions. However, the extracted information does not extend beyond detecting driving lanes.

3) Semantic Segmentation: Semantic segmentation solves a structured pixel-wise labelling problem over meaningful objects in the scene. In early research, maximum-a-posteriori inference in a conditional random field (CRF) [25] was used to compute the labelling layout. More recently, researchers started exploiting deep networks for modelling and extracting these latent feature hierarchies with Fully Convolutional Networks (FCNs) [26]. To improve the output resolution, which suffers from the down- and upsampling in the encoder and decoder path, several solutions have been proposed such as skip connections [9], dilated convolutions [27], and end-to-end integration of a CRF [28].

4) Automatic Label Generation: To fully exploit scene context, the aforementioned networks require large-scale semantic datasets [5]. To reduce the labeling effort for such datasets, several automatic annotation solutions have been proposed. In [29] a single 3D scene annotation is projected

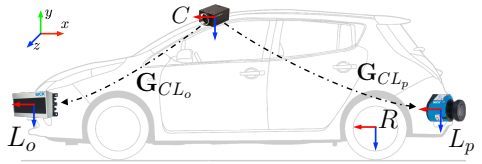


Fig. 2. The vehicle's reference frame R is located at the middle of the rear axle. The approximate sensor locations are shown for the monocular camera C, pushbroom LiDAR L_p, and object detection LiDAR L_o.

into multiple 2D images. The methods proposed in [30], [31] create weakly-supervised annotations for training networks for applications which require less detail and are sometimes supported by a small, manually annotated dataset as in [32]. In this work we automatically create road marking annotations from multimodal data.

III. Weakly-Supervised Annotations from Multimodal Data

We present a method for creating road marking annotations in a weakly-supervised way by leveraging complementary sensor modalities. After the network is trained using these annotations, it requires only a monocular camera at run time. The annotations are computed offline and thus do not require real-time generation.

We exploit the property that road markings are highly reflective and must lie on the road surface. We utilize a LiDAR to capture a point cloud of the environment, with a range and reflectance value associated with each point. The latter is not prone to varying lighting conditions, and thus provides benefits over using (only) camera images. The road surface is extracted from the point cloud using a surface normal region-growing approach and projected into the image to decrease the search area for road markings. A dense CRF is then employed to identify the road marking image pixels by corresponding them with the high-reflectance laser points.

A. Extracting the Road Surface

As road markings only occur on the road itself, coarse segmentation of the road surface can decrease the search domain. This speeds up the algorithm and makes it less prone to false detections (i.e. high-reflectance objects such as white vehicles).

A training route is segmented in $25\,\mathrm{m}$ chunks of laser and image data. The normal of every laser point is calculated using a local neighborhood (empirical evaluation showed that a radius of $0.35\,\mathrm{m}$ achieved good results). From these, the surface normal for the selected point is calculated using principal component analysis (PCA). We employ a per scan-line based region-growing approach (we build our point clouds with a LiDAR mounted in push broom configuration, see Fig. 2) starting at the position of the vehicle and going outwards. The boundary of the road surface is found whenever the surface normal is not parallel to the z-axis of the vehicle anymore. The road surface point cloud is then projected into the camera image using the extrinsic transform $\mathbf{G}_{CL_p}$ to extract the pixels belonging to the road.

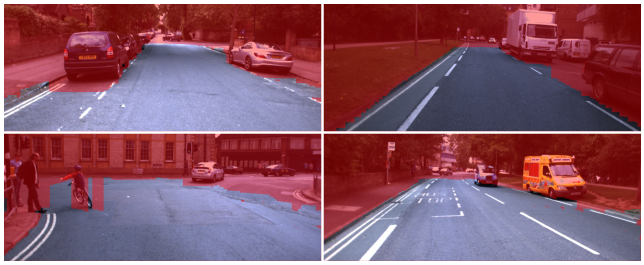


Fig. 3. Four examples of extracted road surfaces generated by the surface normal region-growing approach and object detection mask. Highly accurate results are not necessary as this step is only used to restrict the search domain for later steps.

Since LiDAR L_p is mounted in a push broom configuration (see Fig. 2), at any given time, the fields-of-view of LiDAR L_p and camera C do not overlap. Sensor covisibility is simulated by integrating vehicle egomotion estimates. Thus, and since urban scenes are dynamic, the extracted road surface points can project onto dynamic objects such as cars, cyclists, etc. in the image. Hence, we use an additional horizontal LiDAR L_o on the front of the vehicle to capture static and dynamic objects in the scene, and implement the "stixels"-inspired approach of [30] to remove objects from the extracted road region. In Fig. 3 four examples of extracted road surfaces are shown.

B. Classifying the Road Marking Pixels

After the road surface image pixels are extracted, each pixel should be classified as either *road marking* or *non-road marking*. This is a difficult classification problem, since the non-road marking class has a diverse color and texture domain. We use a CRF to associate image pixels with the high-reflectance points of the sparse laser point cloud, because this is a state-of-the-art method for contextual coherent image segmentation in the presence of prior knowledge (i.e. reflectance values).

A CRF models pixel labels as random variables in an undirected graphical model given some observations (i.e. the image). The labelling task is then posed as an energy minimization problem. The framework of [25] is utilized, in which each pixel is represented by a vertex of the graph, and all vertices are connected to each other by Gaussian edge potentials. These pairwise potentials take into account long-range interactions between pixels. At the same time, they ensure that the mean field approximation of the CRF can be computed in a highly efficient manner, so that optimization over a dense pixel-wise model can be performed within seconds.

Let $X_i \in \mathbf{X}$ be the random variable, which represents the label assigned to pixel $i = \{1, \ldots, N\}$, where N is the number of pixels. Each pixel takes a value in the label space $\mathcal{L} = \{l_r, l_n\}$, where l_r denotes the class *road marking* and l_n denotes the class *non-road marking*. Let $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ be the undirected graph, whose vertices X_i are contained in $\mathcal{V}$ and whose edges are contained in $\mathcal{E}$. Given the graph, the combination of the observed image pixels $\mathbf{I}$ and the label

configuration $\mathbf{X}$ can be modelled as a CRF characterized by the Gibbs distribution

$$p(\mathbf{X} = \mathbf{x}|\mathbf{I}) = \frac{1}{Z(\mathbf{I})} \exp\left(-E\left(\mathbf{x}|\mathbf{I}\right)\right), \tag{1}$$

where $Z(\mathbf{I})$ is the normalization constant and $E\left(\mathbf{x}|\mathbf{I}\right)$ is the Gibbs energy function defined as

$$E\left(\mathbf{x}|\mathbf{I}\right) = \sum_{c \in \mathcal{C}_\mathcal{G}} \Phi_c(\mathbf{x}_c|\mathbf{I}). \tag{2}$$

In (2), $\mathcal{C}_\mathcal{G}$ denotes the set of cliques associated with $\mathcal{G}$, in which each clique c induces a potential Φ_c. The most probable label assignment given the observed image data is thus found by minimizing the energy: $\mathbf{x}^* = \arg\max_{\mathbf{x} \in \mathcal{L}^N} p(\mathbf{X} = \mathbf{x}|\mathbf{I})$. Omitting the conditioning on $\mathbf{I}$ for notational convenience, we use the energy function

$$E\left(\mathbf{x}\right) = \sum_{i \in \mathcal{V}} \psi_i(x_i) + \sum_{(i,j) \in \mathcal{E}} \psi_{ij}(x_i, x_j), \tag{3}$$

where $\psi_i(x_i) : \mathcal{L} \rightarrow \mathbb{R}$ are the unary potentials that denote the cost of pixel i taking label x_i, and $\psi_{ij}(x_i, x_j) : \mathcal{L} \times \mathcal{L} \rightarrow \mathbb{R}$ are the pairwise potentials that denote the cost of assigning the labels x_i and x_j to pixel i and j simultaneously. The unary potential can thus be seen as an independent, discriminative pixel classifier, whereas the pairwise potentials are smoothing terms that encourage similar labels for pixels with similar features.

1) Unary Potentials: Ideally, the measured reflectance value provides a good feature for pixel-wise road marking classification, because we have ensured that the search domain only contains the road surface. Unfortunately, this simple classifier will not give satisfactory results for two reasons.

Firstly, the measured reflectance value is a function of the material, the viewing angle, and the distance of the object. We perform a two-step procedure on a per-beam basis to make the reflectance values of a scene comparable: 1) subtract the per-beam median reflectance value, calculated over the entire dataset, from that beam (since in most cases it will not hit a road marking), 2) normalize the values of that beam by dividing them by the per-beam variance calculated over the entire dataset.

Secondly, a point cloud is significantly sparser than an image. In order to compute a unary potential for every vertex (i.e. pixel), the reflectance values of the point cloud are interpolated linearly. This results in a smooth synthetic laser image (see Fig. 4), which cannot be used for creating pixel-accurate unary potentials. Under the assumption that there exists a correlation between the reflectance and brightness of road marking pixels, a simple solution is to multiply the grayscale pixel intensities g_i with the reflectance values of the synthetic image r_i

$$\psi_i(x_i) = g_i \cdot r_i(x_i). \tag{4}$$

In this way, color and reflectance form a joint, discriminative feature for road marking pixels given the road surface, so that only bright *and* highly reflective pixels are assigned an increased potential.

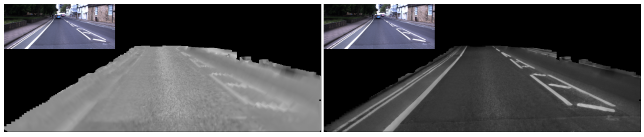


Fig. 4. Generating the unary potentials for the CRF. Interpolating the laser reflectance values results in a smooth synthetic image (*left*) not sufficient for the task. The potentials can be improved by multiplying them with the grayscale intensities of the original image (*right*).

2) Pairwise Potentials: In order to ensure efficient optimization as in [25], define the Gaussian edge potentials as

$$\psi_{ij}(x_i, x_j) = \mu(x_i, x_j) \sum_{m=1}^{M} k_m\left(\mathbf{f}_i, \mathbf{f}_j\right) = \mu(x_i, x_j) K(\mathbf{f}_i, \mathbf{f}_j), \tag{5}$$

where each k_m is a Gaussian kernel which takes a feature vector $\mathbf{f}$ from the respective pixel. We take the compatibility function $\mu(x_i, x_j) = [x_i \neq x_j]$. In contrast to [25], we do not weigh the Gaussian kernels, because learning these weights requires ground-truth labels. However, the same two-kernel potentials are used where the feature vectors $\mathbf{f}$ include the pixel RGB values I at the pixel position p

$$K(\mathbf{f}_i, \mathbf{f}_j) = \exp\left(-\frac{\|p_i - p_j\|^2}{2\theta_\alpha} - \frac{\|I_i - I_j\|^2}{2\theta_\beta}\right) + \exp\left(-\frac{\|p_i - p_j\|^2}{2\theta_\gamma}\right). \tag{6}$$

The first exponential function forces nearby pixels with similar features to have the same label, while the second smoothens the results by removing small, isolated regions. The θ parameters control the amount of influence between pixel i and j; increasing θ will increase long-range interactions. We empirically choose $\theta_\alpha = 43$, $\theta_\beta = 9$, and $\theta_\gamma = 3$. This choice was inspired by [29].

C. Annotation Results

Qualitative evaluation of the created annotations demonstrates that high-quality results are achieved, as illustrated in Fig. 5. The current approach does not classify all the road marking pixels in every image perfectly. This happens due to the fact that the method is unsupervised and the dataset contains images with varying lighting conditions and reflectance range. Learning weights for the kernels to adjust to specific images is challenging due to the lack of ground-truth labels. The results might be improved if weights are learned from a relatively small set of manually labelled images.

However, as shown later in Section V, the generated annotations are sufficient for detecting road markings in urban settings under varying conditions. The most likely reason for this is that several regularization techniques incorporated in the network such as dropout and batch normalization prevent overfitting to the imperfect annotations. The best generalized binary segmentation that the network is able to achieve, groups the road marking pixels in one class, since

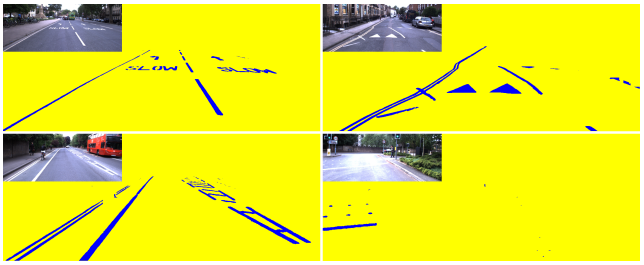


Fig. 5. High-quality annotation results achieved by the CRF approach. Although not all road markings are captured perfectly, these results are sufficient for training. In the case of over-exposure (*bottom right*), annotations are conservatively estimated.

their appearance is very similar to the correctly labelled road marking pixels.

Note that, although the CRF approach achieves good results, it is not suited for real-time applications with the current inference algorithm, because processing of a single high-resolution image takes several seconds. Furthermore, we do not claim that the feature and parameter choices are optimal (see Section III-D), but they generate annotations that are sufficient for training the network, which is the end goal.

D. Alternative Features for the CRF Potentials

We have experimented with different features for the unary and pairwise potentials in order to improve the annotations. Below we briefly share our findings. However, a more extensive analysis is necessary to determine the best overall feature type for this specific application.

For the unary potentials, we found that the Nguyen feature [33] tends to work well in certain settings as a substitute for the grayscale intensities. This is likely because the Nguyen feature emphasizes elongated structures such as lane markings, and is thus less prone to regions of over-exposure.

Intuitively, the RGB values in the pairwise potentials do not seem to be the best feature to discriminate the road surface from road markings, especially not in over-exposed images. Therefore, we have experimented by adding the interpolated reflectance value for every pixel to the feature vector. However, this gave unsatisfactory smoothed results, even when the respective θ value was decreased. Furthermore, we have experimented with different color spaces such as CIELUV and HSV, but empirically achieved the best results across the entire dataset using the RGB values.

IV. Deep Semantic Segmentation Network

Deep neural networks are the state-of-the-art solution for semantic segmentation. We argue that these methods (with adequate training data) will also improve road marking detection and classification, since they are able to leverage the global scene context and are robust to spatial deformations, degradation, and partial occlusion. Besides that, classification is not limited to shapes, but the difference in underlying meaning of similarly shaped road markings (e.g. lane separators and separators that mark a parking spot) can be retrieved based on their place and context in the scene.

A. Network Architecture

We train a U-Net inspired architecture shown in Fig. 6. Like most deep semantic segmentation networks, it consists of an encoding and a decoding path, and a way to provide fine-grained input information to the decoder.

The size of the image is repeatedly reduced by a factor of 2 in the encoder path to increase the receptive field of the filters. Consequently, they become invariant to tiny deformations of the road markings and are able to take contextual information and long-range interactions into account. The decoding path is identical to the encoding path except that the feature maps are now repeatedly upsampled to generate an output image of the same resolution as the input. The upsampling is performed with trainable filters. Skip connections concatenate high-resolution features from the encoding path to the decoding path, so that fast convergence is ensured and a fine-grained segmentation output can be achieved. We modified the original U-Net to include batch normalization after every convolutional filter, and added zero-padding to the sides so that the output resolution is equivalent to the input resolution.

The output of the network is computed by a pixel-wise softmax over the final feature maps

$$p_{i,k} = \frac{\exp(a_{i,k})}{\sum_{m=1}^{M} \exp(a_{i,m})}, \tag{7}$$

where $a_{i,k}$ denotes the activation in feature map k at pixel i, and M is the number of classes. Then, pixel i is assigned a label by $l_i = \arg\max_k p_{i,k}$. Since the number of road marking pixels is much lower than non-road marking pixels, a weighted cross entropy loss is implemented to cope with the class imbalance

$$E = -\sum_{i=1}^{N} w_{l_\star} \log(p_{i,l_\star}), \tag{8}$$

where $l_\star$ is the ground-truth class for that pixel and $w_{l_\star}$ is the weight associated with the ground-truth class of that pixel.

Weights for the two classes are calculated by median frequency balancing $w_m = \tilde{f}/f_m$ [34], where f_m is the total number of pixels of class m divided by the total number of pixels in images where class m is present. The scalar $\tilde{f}$ denotes the median of f_m.

B. Network Training

The parameters that were used during training against the created RobotCar annotations are shown in Table I. We use dropout as a supplementary regularization tool besides batch normalization to prevent overfitting. Training is done from scratch with weight initialization as described in [35].

For the quantitative results, we split up the CamVid dataset into 490 train, 105 validation, and 105 test frames. We select the epoch for testing in which the accuracy is highest among the evaluations on the validation set.

At run time, the TensorFlow implementation of our network in Python performs inference on an input image in 16 ms (=62.5 Hz) using an NVIDIA TITAN Xp GPU.

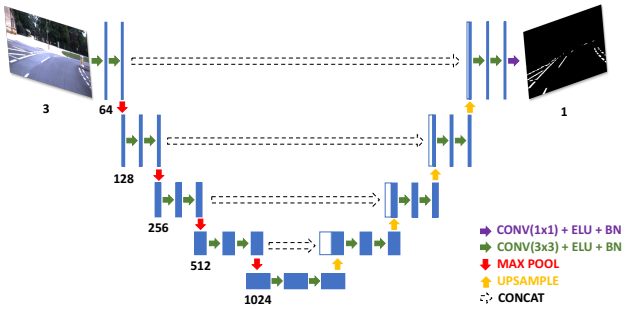


Fig. 6. The U-Net [9] architecture consisting of an encoder and a decoder path, which compresses the feature maps to increase the receptive field of the filters before expanding to a full resolution per-pixel class prediction.

TABLE I
NETWORK & TRAINING PARAMETERS

Network	Value	Training	Value
loss function	weighted cross entropy	batch size	10
activation function	ELU	epochs	100
number of layers	5	optimizer	Adam
filter size	3×3	learning rate	0.0001
max pool size	2	dropout	0.5
stride	1		
image resolution	128 × 320		

V. EXPERIMENTAL RESULTS

Due to the absence of a readily available dataset that contains LiDAR data and pixel-wise ground-truth labels of road markings, we employ the following approach to test our system. We train the network using the weakly-supervised annotations created on the RobotCar (RC) dataset, and then fine-tune with manually created labels on the CamVid (CV) dataset to adapt to the different domain. This process allows for pixel-wise evaluation of our approach against the CamVid labels, which will be used as ground truth. Additionally, we show qualitative performance of the network when trained using only the annotations created on the RobotCar dataset, in a variety of traffic, lighting, and weather conditions.

A. Quantitative Evaluation

We performed five experiments, all tested against the 105 selected ground-truth CamVid labels (see Table II).

The first two experiments depict baseline results on CamVid by training against a small set of, and all available ground-truth labels, respectively. For the remaining experiments, the network was trained using 24238 weakly-supervised RobotCar annotations. Herein, the third experiment was tested directly against the CamVid labels, whereas for the fourth and fifth experiments, the network was fine-tuned on a varying number of CamVid labels. Evaluating the results, the following three key observations can be made:

1) The third experiment clearly illustrates that fine-tuning is necessary. Interestingly, the result demonstrates also that training against a large dataset of another domain outperforms training against a small dataset of the actual test domain. This likely occurs because the network is trained on a bigger variety of traffic and lighting conditions, which improves generalization.

2) The fourth experiment shows that training using the weakly-supervised annotations, while fine-tuning using only 25 manually created CamVid labels, achieves comparable performance (in terms of IoU and F_1) to the baseline result trained on 490 manual labels. This significantly reduces the required labelling effort.

3) The last experiment shows that we outperform the baseline result, when we fine-tune using all available ground-truth labels. This is not trivial, since adding more data from a different domain potentially alters the data distribution. The result indicates that more training data of another domain (which requires no additional labelling effort in our case) improves performance.

Note that the RobotCar annotations were uniquely generated without the use of data augmentation techniques. The results further show that pre-training on the annotations increases the precision but decreases the recall. This is expected, since the annotations are created conservatively (see Fig. 5).

Although the manually created CamVid road marking labels are of high quality, there are instances where the labels do not accurately represent the ground truth. As shown in Fig. 7, the predicted output can then correspond better to the actual ground truth than the label itself. Besides, it is important to keep in mind that object-level performance is more relevant than pixel-wise performance, when road marking detection is performed for planning purposes (which is our future goal).

Fig. 7. The CamVid label (*middle*) and the predicted output (*right*) for a test image. The predicted output reflects the ground truth better at several places in the images such as the lower part of the bounding box around the bicycle and the bicycle itself.

TABLE II
QUANTITATIVE PIXEL-WISE RESULTS ON ROBOTCAR (RC) AND CAMVID (CV) DATASETS

Train Dataset	ACC	PRE	REC	IoU	F_1
25 CV	96.82	46.17	**87.64**	42.03	58.33
490 CV	98.22	63.96	86.33	57.17	71.10
24238 RC	97.92	62.92	65.25	46.17	62.52
24238 RC + 25 CV	98.20	66.39	78.39	54.27	69.54
24238 RC + 490 CV	**98.60**	**72.64**	81.63	**61.20**	**75.04**

B. Qualitative Evaluation

Fig. 8 shows qualitative results on a RobotCar test dataset. The results demonstrate that the network segments the road markings from the image without any preprocessing steps when trained using the weakly-supervised annotations. Even

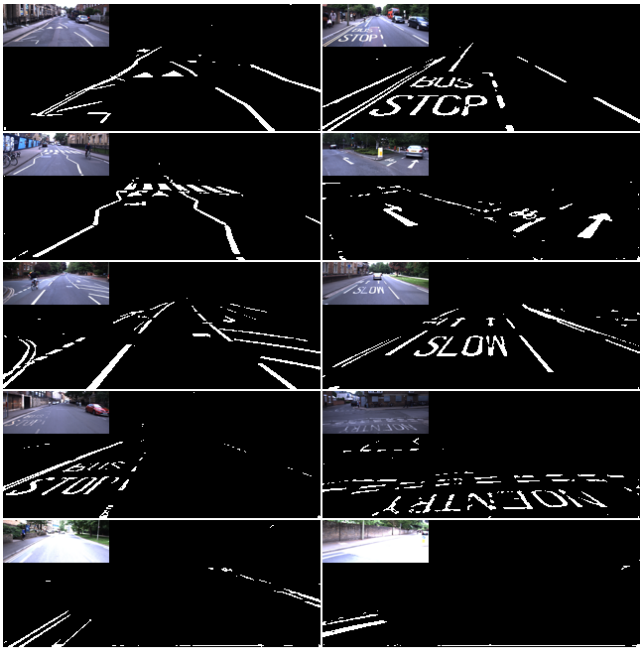


Fig. 8. Network output on RobotCar images when trained against weakly-supervised annotations. The network accurately detects all road markings without limitations to shape, even when the road markings start to degrade (*fourth row*). In case of over-exposure (*fifth row*), a conservative segmentation is achieved.

in case of degradation, the network is able to sufficiently segment the road markings. The network achieves a conservative segmentation in cases of over-exposure, where intensity-based approaches most likely fail.

Additionally, we trained a network using annotations generated under different lighting and weather conditions. Fig. 9 shows the network output under these conditions at the same location. Although the method performs best in overcast conditions, the results under more difficult conditions appear satisfactory considering the image quality.

C. Limitations

Under some conditions the quality of the annotation is poor, as illustrated in Fig. 10. Bright parts of the pavement can be mistaken for road marking, when the extraction algorithm has difficulties finding the correct road surface border. These failure cases could be addressed by more complex road extraction algorithms, or a more discriminative (supervised) feature set for the CRF potentials. These annotations were not included in the training set.

Furthermore, the network output can be spurious at times in the presence of parked cars or stark shadow lines, as shown in Fig. 11. False detections occur, because object edges introduce high-intensity gradients at the same place in the image where road markings normally appear. This can likely be resolved when the network is given annotations with road marking types, so that it can learn improved spatial and contextual coherence.

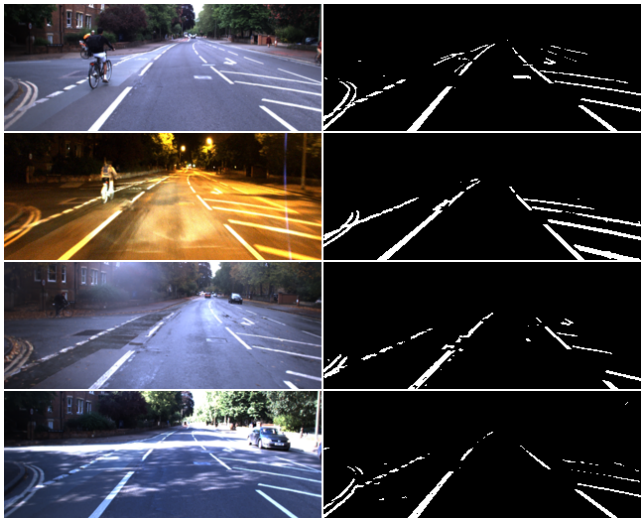


Fig. 9. Road marking detection under different conditions (overcast, night, rain, and sun) at the same location. Despite significant changes in appearance, the method achieves satisfactory results.

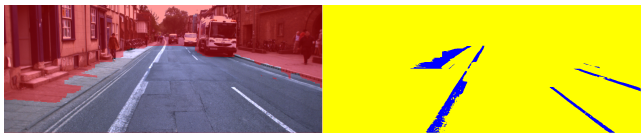


Fig. 10. Poor quality annotation due to insufficient road extraction, because the pavement is approximately at road height. The result can be improved by more accurate road extraction algorithms or a more discriminative feature set for the CRF potentials.

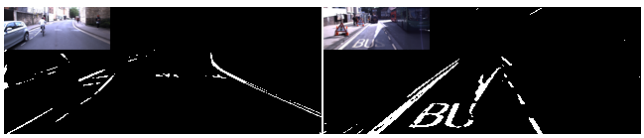


Fig. 11. Examples of spurious network output in the presence of parked cars and stark shadow lines, because edges introduce high-intensity gradients at the same place in the image where road markings normally appear.

VI. Conclusion

We have presented a weakly-supervised system for real-time road marking detection using images of complex urban environments obtained from a monocular camera. At run time, the road markings in the scene are detected using a deep segmentation network without relying on any pre-processing step or predefined models. Crucially, by leveraging LiDAR reflectance values in a CRF approach, we generated vast quantities of annotated road marking images for training purposes in a weakly-supervised way, thereby avoiding the need for expensive manual labelling. We have demonstrated reliable qualitative performance under varying traffic, lighting, and weather conditions on the Oxford RobotCar dataset. Furthermore, we showed quantitatively on the CamVid dataset that weakly-supervised annotations of another domain significantly reduce the required labelling effort and improve performance.

In future work we will extend the current framework to

include semantic classification of the road markings in the scene to retrieve the rules of the road. This information will be exploited to aid high-level scene understanding, mapping, and planning in complex urban environments.

REFERENCES

[1] H. Xu, Y. Gao, F. Yu, and T. Darrell, "End-to-end learning of driving models from large-scale video datasets," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2174–2182.

[2] M. Schreiber, C. Knöppel, and U. Franke, "Laneloc: Lane marking based localization using highly accurate maps," in *Intelligent Vehicles Symposium (IV), 2013 IEEE*. IEEE, 2013, pp. 449–454.

[3] M. Schreiber, F. Poggenhans, and C. Stiller, "Detecting symbols on road surface for mapping and localization using ocr," in *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*. IEEE, 2014, pp. 597–602.

[4] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[5] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3213–3223.

[6] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 Year, 1000km: The Oxford RobotCar Dataset," *The International Journal of Robotics Research (IJRR)*, vol. 36, no. 1, pp. 3–15, 2017.

[7] O. Bailo, S. Lee, F. Rameau, J. S. Yoon, and I. S. Kweon, "Robust road marking detection and recognition using density-based grouping and machine learning techniques," in *Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on*. IEEE, 2017, pp. 760–768.

[8] T. Ahmad, D. Ilstrup, E. Emami, and G. Bebis, "Symbolic road marking recognition using convolutional neural networks," in *Intelligent Vehicles Symposium (IV), 2017 IEEE*. IEEE, 2017, pp. 1428–1433.

[9] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2015, pp. 234–241.

[10] G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognition Letters*, vol. 30, no. 2, pp. 88–97, 2009.

[11] T. Veit, J.-P. Tarel, P. Nicolle, and P. Charbonnier, "Evaluation of road marking feature extraction," in *Intelligent Transportation Systems, 2008. ITSC 2008. 11th International IEEE Conference on*. IEEE, 2008, pp. 174–181.

[12] T. Wu and A. Ranganathan, "A practical system for road marking detection and recognition," in *Intelligent Vehicles Symposium (IV), 2012 IEEE*. IEEE, 2012, pp. 25–30.

[13] D. Hyeon, S. Lee, S. Jung, S.-W. Kim, and S.-W. Seo, "Robust road marking detection using convex grouping method in around-view monitoring system," in *Intelligent Vehicles Symposium (IV), 2016 IEEE*. IEEE, 2016, pp. 1004–1009.

[14] B. Qin, W. Liu, X. Shen, Z. J. Chong, T. Bandyopadhyay, M. Ang, E. Frazzoli, and D. Rus, "A general framework for road marking detection and analysis," in *Intelligent Transportation Systems-(ITSC), 2013 16th International IEEE Conference on*. IEEE, 2013, pp. 619–625.

[15] F. Poggenhans, M. Schreiber, and C. Stiller, "A universal approach to detect and classify road surface markings," in *Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on*. IEEE, 2015, pp. 1915–1921.

[16] J. Greenhalgh and M. Mirmehdi, "Automatic detection and recognition of symbols and text on the road surface," in *International Conference on Pattern Recognition Applications and Methods*. Springer, Cham, 2015, pp. 124–140.

[17] B. Mathibela, P. Newman, and I. Posner, "Reading the road: road marking classification and interpretation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 4, pp. 2072–2081, 2015.

[18] T. Woudsma, L. Hazelhoff, P. H. de With, and I. Creusen, "Automated generation of road marking maps from street-level panoramic images," in *Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on*. IEEE, 2015, pp. 925–930.

[19] T. Chen, Z. Chen, Q. Shi, and X. Huang, "Road marking detection and classification using machine learning algorithms," in *Intelligent Vehicles Symposium (IV), 2015 IEEE*. IEEE, 2015, pp. 617–621.

[20] M. Cheng, H. Zhang, C. Wang, and J. Li, "Extraction and classification of road markings using mobile laser scanning point clouds," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 3, pp. 1182–1196, 2017.

[21] Y. Yu, J. Li, H. Guan, F. Jia, and C. Wang, "Learning hierarchical features for automated extraction of road markings from 3-d mobile lidar point clouds," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 8, no. 2, pp. 709–726, 2015.

[22] A. Bar Hillel, R. Lerner, D. Levi, and G. Raz, "Recent progress in road and lane detection: A survey," *Mach. Vision Appl.*, vol. 25, no. 3, pp. 727–745, Apr. 2014.

[23] B. Huval, T. Wang, S. Tandon, J. Kiske, W. Song, J. Pazhayampallil, M. Andriluka, P. Rajpurkar, T. Migimatsu, R. Cheng-Yue, F. Mujica, A. Coates, and A. Y. Ng, "An empirical evaluation of deep learning on highway driving," *CoRR*, vol. abs/1504.01716, 2015.

[24] B. He, R. Ai, Y. Yan, and X. Lang, "Lane marking detection based on convolution neural network from point clouds," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, Nov 2016, pp. 2475–2480.

[25] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected CRFs with gaussian edge potentials," in *Advances in Neural Information Processing Systems 24*, J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2011, pp. 109–117.

[26] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, and J. Garcia-Rodriguez, "A review on deep learning techniques applied to semantic segmentation," *arXiv preprint arXiv:1704.06857*, 2017.

[27] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs." *IEEE transactions on pattern analysis and machine intelligence*, 2017.

[28] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. Torr, "Conditional random fields as recurrent neural networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 1529–1537.

[29] J. Xie, M. Kiefel, M.-T. Sun, and A. Geiger, "Semantic instance annotation of street scenes by 3d to 2d label transfer," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3688–3697.

[30] D. Barnes, W. Maddern, and I. Posner, "Find your own way: Weakly-supervised segmentation of path proposals for urban autonomy," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 203–210.

[31] J. M. Alvarez, T. Gevers, Y. LeCun, and A. M. Lopez, "Road scene segmentation from a single image," in *European Conference on Computer Vision*. Springer, 2012, pp. 376–389.

[32] W. Wang, N. Wang, X. Wu, S. You, and U. Neumann, "Self-paced cross-modality transfer learning for efficient road segmentation," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 1394–1401.

[33] U. T. Nguyen, A. Bhuiyan, L. A. Park, and K. Ramamohanarao, "An effective retinal blood vessel segmentation method using multi-scale line detection," *Pattern recognition*, vol. 46, no. 3, pp. 703–715, 2013.

[34] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2650–2658.

[35] K. He, X. Zhang, S. Ren, and J. 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.

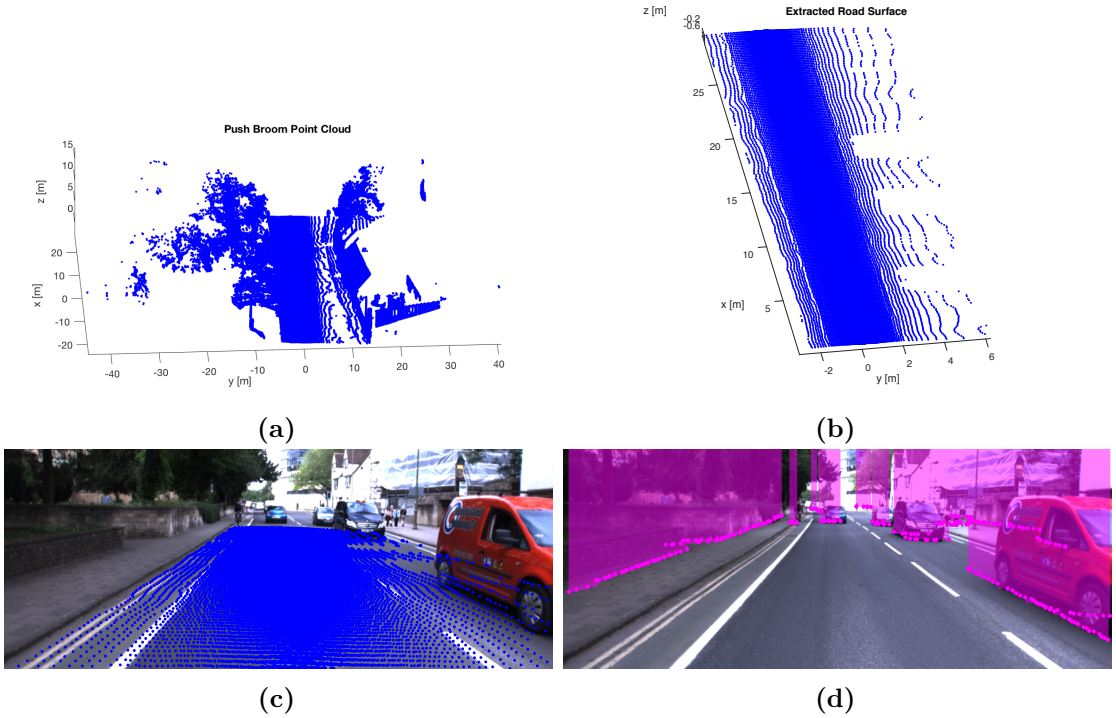


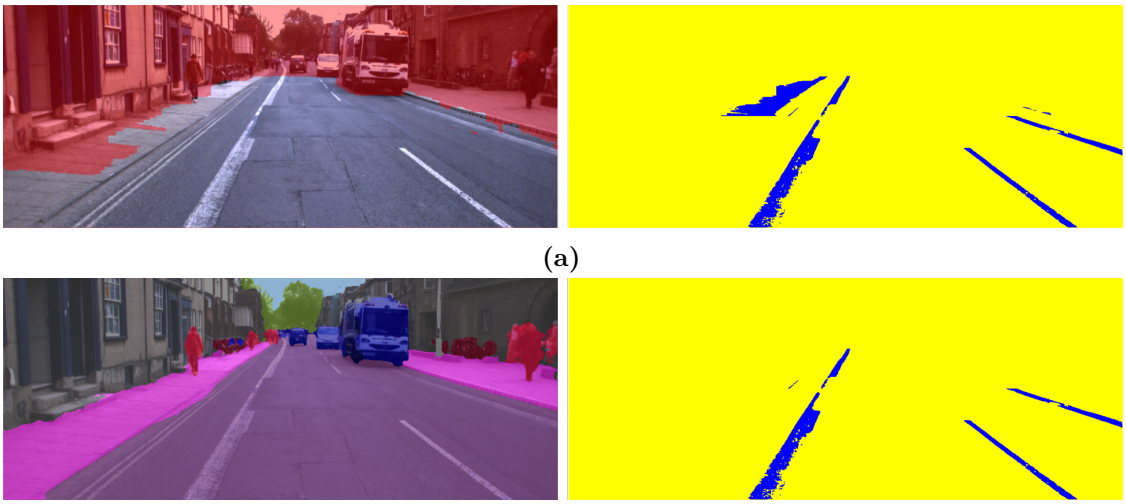
Figure 5.3: The substeps of the road surface extraction algorithm. A surface-normal region-growing approach extracts the points of the road surface, **(b)**, from a chunk of push-broom LiDAR data (displayed in bird's-eye view), **(a)**. As the fields-of-view of LiDAR L_p and camera C do not overlap at any given time, points of the extracted road surface can be projected onto dynamic objects, **(c)**. Therefore, LiDAR L_o computes an object mask, **(d)**, which is combined with the road surface point cloud to extract the road surface pixels.

5.2 Approximated Road Marking Labels

This section expands upon the material presented in Section III of the reproduced publication by providing additional implementation details, further qualitative results, and a discussion regarding the automatic generation of the approximated road marking labels.

5.2.1 Further Details

This section contains further details regarding the road surface extraction and the use of LiDAR reflectance values.



(a)

(b)

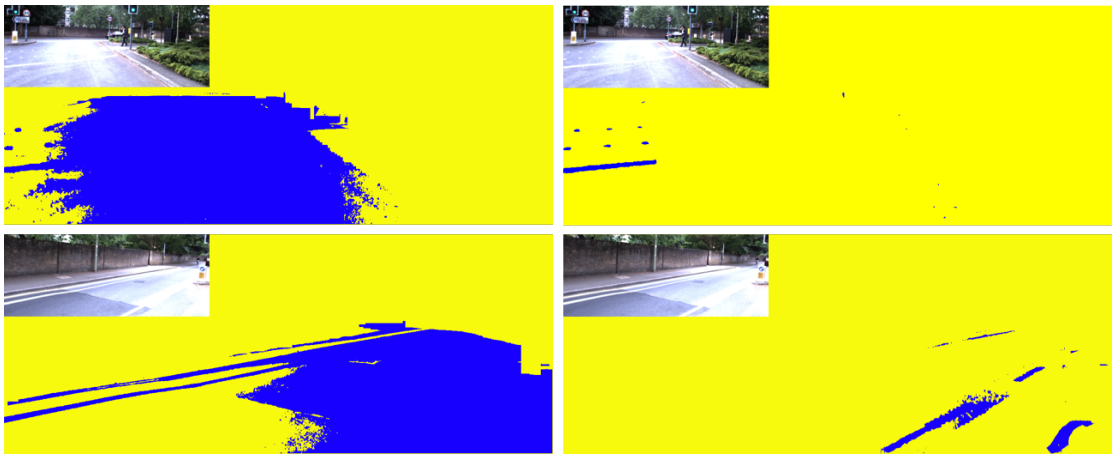
Figure 5.4: Approximated labels can be of poor quality, as shown in the reproduced publication, due to inaccurate road surface extraction when the pavement is approximately at the road height, **(a)**. Because of improvements in semantic segmentation, this issue is now resolved by obtaining the road mask of a DNN trained on the Cityscapes dataset, **(b)**.

Road Surface Extraction

The various substeps of the road extraction algorithm are visualized in Figure 5.3. As shown in the publication, the current algorithm can lead to poor-quality approximated labels in the presence of low curbs and bright pavements. These failure cases can be addressed by employing a better performing method for the road surface extraction. For instance, the image can be segmented semantically by a DNN trained on Cityscapes data (similar to Section 6.2), thereby directly obtaining a pixel-wise road surface mask while using less computational time. The two approaches are compared for an example scene in Figure 5.4.

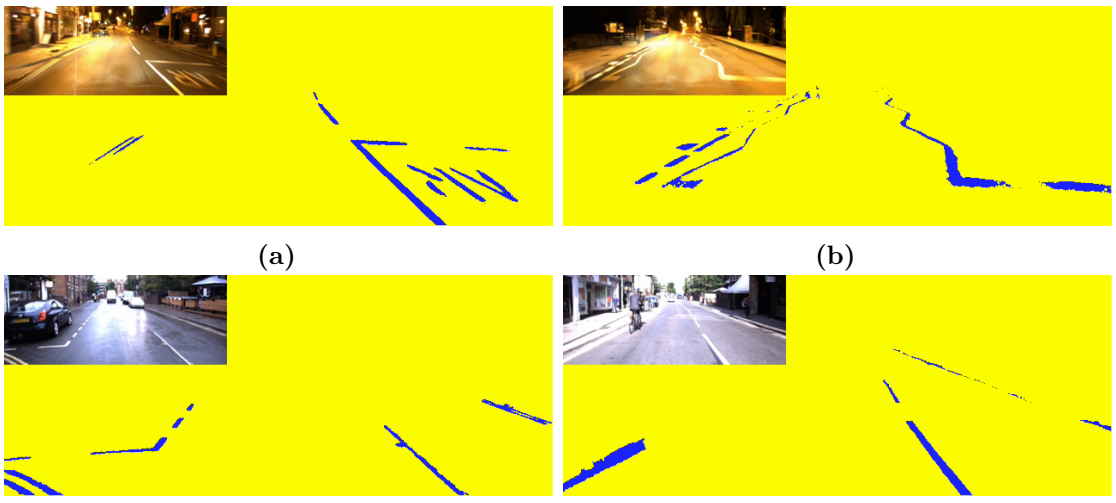
LiDAR Reflectance

Figure 5.5 demonstrates the advantage of using LiDAR reflectance values for the unary potentials of two overexposed scenes. The CRF is able to classify only the road markings pixels instead of all the bright pixels on the road surface because the LiDAR reflectance is not affected by lighting.



(a) (b)

Figure 5.5: LiDAR reflectance is a suitable complimentary discriminating feature besides the pixel values as it is not affected by lighting. The CRF optimization leads to drastic errors for overexposed scenes when only pixel values are used for the features, **(a)**, as compared to the conservative classification achieved when the LiDAR reflectance values are also taken into account, **(b)**.



(a) (b)

(c) (d)

Figure 5.6: Generated approximated road marking labels during nighttime, **(a)** & **(b)**, wet conditions, **(c)**, and sunny conditions, **(d)**. The LiDAR reflectance values are not affected by lighting and thus allow for a conservatively generated label, which is sufficient for training a DNN when adequate regularization is implemented.

5.2.2 Further Results

Qualitative results of the generated approximated road marking labels under different conditions are shown in Figure 5.6. The LiDAR reflectance values are crucial in these cases as large parts of the road surface are overexposed due to either the headlights

of the ego vehicle, wetness, or sun. The labels are conservatively generated to favour precision over recall. Nevertheless, they are sufficient for training a DNN for road marking segmentation, as demonstrated in the reproduced publication.

The reader is referred to the video[2] accompanying the publication for additional qualitative results.

5.2.3 Further Discussion

Due to the unsupervised nature of the proposed approach, it is ultimately infeasible to generate a perfect ground-truth label for every image as the lighting conditions and reflectance values vary significantly across the dataset. The results might be improved if kernel weights for the CRF are learned from a relatively small set of manually-labelled images, but these were not available in our case.

Moreover, improvements in image-to-image translation [94] have made it possible to model realistic appearance changes at high-resolution. These methods only alter the appearance of the image while keeping the structure and geometry of the scene intact. Therefore, an approximated label generated under overcast conditions can be paired with a synthesized nighttime image. This exact principle is leveraged in Chapter 3 and is a state-of-the-art solution for self-supervised scene understanding under varying conditions. Such methods will likely outperform the method presented in the paper when generating approximated labels in challenging weather and lighting conditions.

Although various aspects could improve the quality of the approximated labels, it is currently unclear how those will affect the end quality of the road marking segmentation. As will become apparent throughout this thesis, there exists a complex interaction between the quality of the approximated labels and the network optimization, which determines the final segmentation performance. In Section 5.3.1, we show that regularization prevents the network from overfitting to the approximated labels.

[2] `https://www.youtube.com/watch?v=2vR00jDYgTI`

Table 5.1: The effect of varying the percentage of dropout for road marking segmentation when trained on approximated labels.

Dropout	Scene A	Scene B	Scene C	Scene D
				
75%				
50%				
25%				
0%				

5.3 DNN Road Marking Segmentation

This section expands upon the material presented in Section IV and V of the reproduced publication by providing additional implementation details, further results, and a discussion regarding road marking segmentation with approximated labels.

5.3.1 Further Details

As mentioned in the publication, including dropout in the DNN is crucial because it allows for correct regularization. Concretely, it allows the network to extend its prediction towards road marking pixels that were excluded from the approximated labels since they share more similarities with the road marking class than the background class. We investigate qualitatively how different dropout values affect the quality of the road marking segmentation in Table 5.1. In the case of no dropout (0%) the predictions are similar to the approximated labels (i.e. high precision, but low recall), which is insufficient as input for scene understanding frameworks. As the percentage of dropout is increased, the recall increases up to the point at which all road marking pixels are segmented, but at the cost of losing precision. An optimal point could be determined by tuning over a small manually-labelled ground-truth dataset.

5.3.2 Further Results

We expand upon the quantitative results presented in the reproduced publication with two additional experiments. Phenomena such as domain adaptation, transfer learning, and catastrophic forgetting [95] play a significant role in both of these experiments. We would like to point out that these experiments are merely performed to provide quantitative insights in the absence of Oxford RobotCar (RC) ground-truth road marking labels; a more thorough investigation of these phenomena in this context is outside the scope of this thesis.

The reader is referred to the video[3] accompanying the publication for additional qualitative results.

Experimental Setup

In the first experiment, we pretrain the network on a large number of approximated labels from the RC dataset and then fine-tune and test with ground-truth labels from either the CamVid (CV) dataset or the Rainy RobotCar (RR) dataset (Appendix B, only the clear images) to adapt to the new domain. This experiment is similar to the one presented in the reproduced publication except that we have changed the CamVid train/validation/test split according to [96] and set dropout to 0.75. Furthermore, we perform an ablation study to compare pretraining on the Cityscapes (CS) dataset with pretraining on the approximated labels. In the second experiment, we add approximated labels directly to the Rainy RobotCar labels and train from scratch on the combined dataset.

For the CV dataset, 376 train, 101 validation, and 232 test images are provided. The RR dataset was split into 303 train, 100 validation, and 100 test (clear) images. For the CS experiments, we pretrain using the provided training set of 2975 images. For every experiment, we train until convergence with the parameters as listed in the publication (except for dropout being set to 0.75) and select the epoch for testing in which the F_1 score is highest among the evaluations on the validation set.

[3] `https://www.youtube.com/watch?v=2vR00jDYgTI`

Experiment 1: Pretraining on Approximated Road Marking Labels

We train five models for each domain, all of which are validated and tested against the CV/RR ground-truth labels. Models 2 and 4 depict baseline results in which we train either against a small set (i.e. 25) of CV/RR ground-truth labels or all available ones, respectively. The other models are pretrained on a large set (i.e 24238) of the approximated RC road marking labels. Model 1 is tested directly against the ground-truth labels, whereas the network is fine-tuned on a varying number of ground-truth labels for models 3 and 5. The following observations are made by evaluating the results in Table 5.2 (with IoU as the main metric for semantic segmentation):

- Model 1 clearly illustrates that there is an apparent domain gap between the approximated labels and the two test datasets, making fine-tuning is necessary.

- Model 3 shows that pretraining on the approximated labels captures valuable knowledge about road markings within the network, which generalizes to other domains and cannot be obtained when only a small number (i.e. 25) of ground-truth labels are available in the domain of interest. Crucially, the performance is boosted without additional manual labelling.

- Model 5 shows that the performance is also boosted by pretraining when we fine-tune using all available ground-truth labels.

- Comparing the baseline and fine-tuned models, it is clear that pretraining on the approximated labels increases the precision but decreases the recall. This is expected as the approximated labels are created conservatively, often favouring precision over recall, as mentioned in the publication.

- The RR models outperform the CV models likely due to two reasons. Firstly, the CV road marking labels are of lesser quality. There are instances where the manual labels do not accurately represent the ground truth. As shown in the publication, the predicted output is then likely to correspond better to the actual ground truth than the label itself. Secondly, the CV dataset has a

Table 5.2: Pixel-wise road marking segmentation results when pretrained on RobotCar approximated labels and fine-tuned on either CamVid or Rainy RobotCar ground-truth labels before testing.

			CamVid				Rainy RobotCar (clear)			
Model	**(Pre)Train Set**	**Fine-Tune Set**	**Pre**	**Rec**	**F_1**	**IoU**	**Pre**	**Rec**	**F_1**	**IoU**
(1)	24238 RC		62.46	60.55	59.16	43.22	62.44	52.47	55.29	39.78
(2)	25 CV/RR		54.81	79.17	62.77	46.72	64.93	80.35	70.21	55.73
(3)	24238 RC	25 CV/RR	61.65	76.96	66.18	50.65	64.85	81.97	70.82	56.40
(4)	376 CV/303 RR		64.98	**83.82**	71.10	57.09	68.85	**90.78**	77.29	64.31
(5)	24238 RC	376 CV/303 RR	**66.08**	83.42	**72.33**	**57.82**	**74.61**	85.50	**78.75**	**66.28**

Table 5.3: Pixel-wise road marking segmentation results when pretrained on either approximated RobotCar labels or Cityscapes labels and fine-tuned on CamVid ground-truth labels before testing.

			CamVid			
Model	**(Pre)Train Set**	**Fine-Tune Set**	**Pre**	**Rec**	**F_1**	**IoU**
(6)	2974 RC		64.25	42.96	48.80	34.64
(7)	2974 RC	25 CV	59.09	78.22	65.32	49.55
(3)	24238 RC	25 CV	61.65	76.96	66.18	50.65
(9)	2975 CS	25 CV	50.83	79.77	59.51	43.71
(10)	2974 RC	376 CV	60.94	**86.30**	70.01	55.00
(5)	24238 RC	376 CV	**66.08**	83.42	**72.33**	**57.82**
(11)	2975 CS	376 CV	62.14	84.40	70.03	55.20

more extensive domain shift within the dataset itself because it consists of sequences collected under different lighting conditions and camera orientation, whereas the RR dataset is captured under much more consistent conditions.

Furthermore, we have performed an ablation study to investigate the effect of pretraining on the approximated labels by comparing it against pretraining on the urban semantic segmentation labels of Cityscapes. We have pretrained with approximately the same number of labels and with equivalent settings to ensure a fair comparison. As the number of classes in Cityscapes is different from the binary road marking segmentation, the final layer was initialized randomly, and the network cannot be tested directly (i.e. without fine-tuning). The following insights are observed from the results in Table 5.3:

- Models 7 and 9 show that it is beneficial to pretrain with the approximated labels instead of CS labels whenever only a small number of manual labels are available for fine-tuning. Intuitively this makes sense as the features of the network are optimized to segment larger objects, which are visually and geometrically different from road markings, in the case of pretraining on Cityscapes, and there are not enough manual labels available to fine-tune these features accordingly.

- Models 10 and 11 show that the difference between pretraining on CS or an equivalent number of approximated labels is negligible whenever a larger number of manual labels are available for fine-tuning. However, neither experiment achieves better performance than model 4, where we train on the full CamVid dataset from scratch. Although this is not trivially explainable, it may indicate that the knowledge captured in pretraining is actually available in the full dataset itself and some negative transfer [95] may be occurring due to the domain differences.

- Models 5 and 11 show that it is beneficial compared to pretraining on CS whenever we pretrain on a larger number of approximated labels. This is still a fair comparison as the approximated labels are generated automatically without requiring additional manual labelling. Similarly, models 1 and 6 show that training on a larger number of approximated labels captures more valuable domain knowledge.

Experiment 2: Adding Approximated Road Marking Labels to the Training Set

In the second experiment, we add approximated labels directly to the RR dataset and train from scratch. We choose to add a relatively small number of approximated labels as we empirically observed that adding a larger number deteriorates performance due to the domain differences and the approximated labels having lower recall than the ground-truth labels. A potential workaround for this problem was recently published [97]. However, that, as well as finding the optimal number of

Table 5.4: Pixel-wise road marking segmentation results when approximated RobotCar labels are added to the ground-truth Rainy RobotCar labels.

		Rainy RobotCar (clear)			
Model	**Dataset**	**Pre**	**Rec**	**F_1**	**IoU**
(1)	24238 RC	62.44	52.47	55.29	39.78
(2)	25 RR	64.93	80.35	70.21	55.73
(12)	25 RR + 25 RC	67.42	80.57	71.47	57.13
(4)	303 RR	68.85	**90.78**	77.29	64.31
(13)	303 RR + 100 RR	**72.87**	88.98	**79.24**	**66.67**

approximated labels to add, is outside the scope of this thesis. It is clear from the results in Table 5.4 that the performance of both model 2 and model 4 can be boosted by adding approximated labels to the dataset. We again observe that model 13 has increased precision but decreased recall, as is the case for most approximated labels.

5.3.3 Further Discussion

Although the presented experiments show that performance boosts are achieved in domains where only a limited number of ground-truth labels are available, these approaches are no longer favoured due to recent improvements in high-resolution image synthesis [68]. As demonstrated in Section 6.2, directly synthesizing data for the domain of interest achieves significantly higher performance gains (for road marking classification) while avoiding the influence of complex learning phenomena due to domain differences. However, there is still a need for binary road marking labels, which the framework presented in this chapter provides at a low cost, since these images are synthesized from semantic maps.

The segmentation quality under adverse conditions could be boosted further by improving the network architecture. For instance, the framework demonstrated in Chapter 3 for semantic segmentation can be applied directly for road marking segmentation as well.

Nevertheless, the required performance of the segmentation network and abstraction level of the output remain an open question in regard to high-level scene understanding. Intuitively, one might think that it is not strictly necessary to detect

every single pixel of a particular road marking to infer its underlying meaning. Alternative approaches [86], [98] draw bounding boxes around road markings, providing a class and an approximate location, which might be sufficient. Although this works well for alphanumerics and signs, it does not suffice for lane structures. Therefore, it is reasonable to assume that pixel-wise segmentation remains relevant in the future as a first step towards semantic understanding.

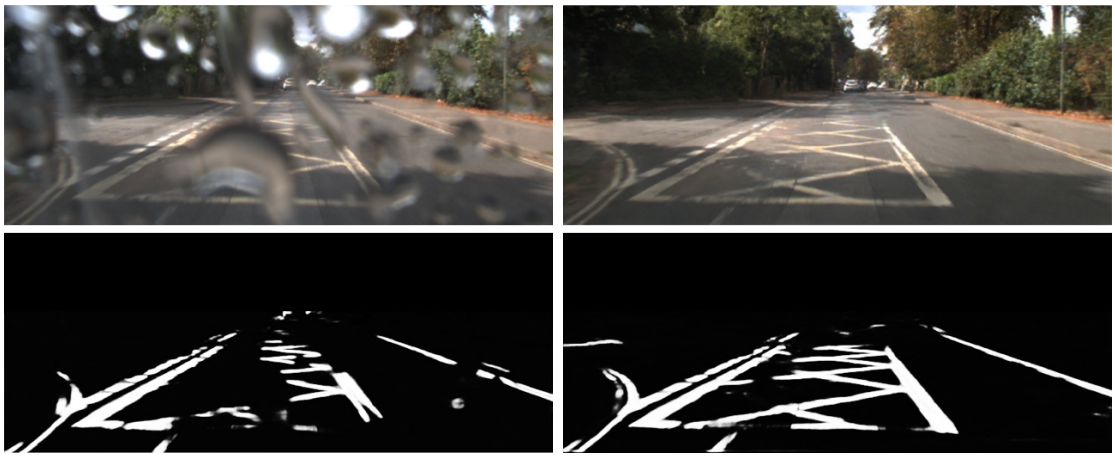
5.4 DNN Road Marking Segmentation under Rainy Conditions

This section extends our investigation of road marking segmentation under challenging weather conditions, as presented in the publication in Section 5.1, by evaluating and demonstrating an approach that reduces the effect of camera lens distortions, which are caused by adherent raindrops and water streaks.

In contrast to the environmental conditions shown in the publication, lens distortions such as adherent raindrops [22] or soil [99] deteriorate the view and consequently the segmentation performance more drastically. In order to correctly segment the road markings under these circumstances, the image is preprocessed (i.e. de-rained) so that the input to the segmentation network is almost equivalent to an overcast image [22]. An example is given in Figure 5.7. This process is summarized in Section 5.4.1 and described in full detail in the publication in Appendix B. Qualitative and quantitative results presented in the publication show that the introduced framework restores the road marking segmentation under the examined conditions.

5.4.1 De-Raining Images

The framework presented in the publication in Appendix B works as an image preprocessor, taking an image with adherent raindrops and streaks of water as input and outputting a clear de-rained image. The output is visually and computationally almost indistinguishable from real clear images. Therefore, accurate road marking segmentation can be achieved using a simple deep semantic segmentation network



(a) **(b)**

Figure 5.7: Raindrops and water streaks adherent to the lens significantly distort the view and consequently the segmentation performance, **(a)**. The image is de-rained by an image-to-image translation network and the road marking segmentation is thereby restored, **(b)**.

trained on overcast conditions, similar to the one described in the publication in Section 5.1.

More concretely, the state-of-the-art image-to-image translation framework Pix2PixHD [68] is trained to learn a mapping from rainy to clear images. The motivation behind this method is that raindrops bend and attenuate the light field in a structured but non-linear way. The DNN learns to model this behaviour and reverses it accordingly.

A stereo rain dataset, consisting of rainy images and their clear counterparts, was recorded by a custom-designed and constructed narrow-baseline camera setup for training and evaluation. Ground-truth road marking labels were manually annotated[4] and made publicly available[5] for 500 images of this dataset.

5.4.2 Summary of the Results

In order to evaluate to what extent road marking segmentation is affected by adherent raindrops and improved by de-raining the images, the publication provides results on the two datasets used in the previous sections. As the CamVid dataset

[4]with thanks to Valentina-Nicoleta Musat

[5]`https://ciumonk.github.io/RobotCar-rainy/`

does not contain any images in rainy conditions, computer-generated raindrops were added to the images. For the Rainy RobotCar dataset, we test against computer-generated rainy images as well as the recorded rainy dataset.

All of the evaluated models were trained similarly to the publication in Section 5.1. Their performance was evaluated for four different cases:

- The clear case represents a model trained on only clear images, which are not affected by rain or preprocessing, and tested against clear images. This serves as a baseline and an upper bound.
- The rainy case represents the same model but now tested against images containing adherent raindrops and water streaks.
- The augmented case represents a model trained on a combination of rainy and clear images, tested against the rainy images.
- The de-rained case represents the clear model again but now tested against rainy images that have been preprocessed by the presented approach.

The following observations are made from the quantitative results presented in the publication:

- The performance severely degrades when a model trained on clear images is tasked to segment road markings in rainy images.
- Retraining the road marking segmentation models against a dataset augmented with rainy images leads to an improvement over the previous case.
- De-raining the images using the presented method before segmenting them performs best among all cases and restores the performance of the segmentation to levels that are close to the baseline.

Re-training the segmentation model with a dataset augmented with rainy images improves performance, as expected. However, using a specialized de-raining preprocessing step significantly outperforms this approach. This is the expected

advantage of having a model dedicated to a specific image-to-image mapping task (i.e. de-raining) in its entirety, as this narrows the variety of images fed to the segmentation task.

5.5 Conclusion

This chapter presented an approach for efficient and scalable road marking segmentation in complex urban environments using images from a monocular camera. The road markings are paramount in representations that allow for navigation and decision making, such as the scene graph demonstrated in Chapter 4.

Crucially, to avoid the need for expensive pixel-wise manual labelling, we learn a binary road marking segmentation first, which opens up the possibility of self-supervised learning. This decision was strengthened by the assumption that the binary segmentation can later be extended towards semantic classification by leveraging additional domain knowledge, as shown in Section 6.1. We have employed LiDAR reflectance values and exploited the fact that road markings are highly reflective to generate vast quantities of approximated road marking labels automatically by optimizing a CRF. However, these approximated labels are not guaranteed to be equivalent to the ground truth because of the automatic nature of this approach. Although there exist methods for improving these approximated labels, two important things should be kept in mind. Firstly, image-to-image translation frameworks have made substantial progress over time. Consequently, the reverse process (i.e. generating an image for a pregenerated label) is now feasible. This approach, presented in Section 6.2, combines segmentation and classification into a single self-supervised learning process. Secondly, the eventual segmentation performance is influenced by a complex interaction of the quality of the approximated labels and the network optimization. Performance gains might be achieved with less effort by improving the latter.

We have qualitatively demonstrated that the approximated labels can be employed to train DNNs for road marking segmentation under different weather and

lighting conditions in various urban traffic situations without relying on preprocessing steps or predefined models. Additionally, we have investigated the performance of the road marking segmentation when adherent raindrops and streaks of water on the camera lens significantly distort the view. An image-to-image translation network was implemented as an image preprocessor, which de-rains the image before segmenting it, thereby restoring the segmentation performance. Moreover, it was shown empirically that dropout functions as a hyperparameter that determines to what extent the network extends its predictions beyond the approximated labels towards segmenting the full set of road marking pixels. Furthermore, we have proven quantitatively that (pre)training on the approximated labels improves the segmentation in different domains where only a limited number of ground-truth labels are available. Notably, all trained networks only require input from a monocular camera during deployment and run online, ensuring that the presented framework is universally deployable.

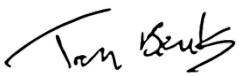
Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor **(only required where there isn't already a statement of contribution within the paper itself).**

Title of Paper	Mark Yourself: Road Marking Segmentation via Weakly-Supervised Annotations from Multimodal Data
Publication Status	Published
Publication Details	**T. Bruls**, W. Maddern, A. A. Morye, and P. Newman, "Mark yourself: Road marking segmentation via weakly-supervised annotations from multimodal data", in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 1863–1870.

Student Confirmation

Student Name:	Tom Adriaan Hubert Bruls		
Contribution to the Paper	All work except editorial changes and advice. Contributions included: - Generating the ideas. - Developing the software. - Preparing and processing the data. - Running the experiments. - Performing the analysis. - Writing the paper, creating the figures and tables. - Presenting the work at the conference.		
Signature		Date	10-05-2020

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Professor Paul Newman			
Supervisor comments			
Signature		Date	11-05-2020

This completed form should be included in the thesis, at the end of the relevant chapter.

6

Representations for Conduct II: Road Marking Classification

Contents

6.1 Model-Driven Road Marking Classification 82
6.2 Data-Driven Road Marking Classification 83
6.2.1 Publication 84
6.2.2 Further Details 93
6.2.3 Further Results 95
6.2.4 Further Discussion 96
6.3 Qualitative Comparison 97
6.4 Conclusion 101

Semantic understanding of the road markings in a traffic scene is a prerequisite for the scene graph as it conveys the *road rules* and thereby provides guidance for decision making. We obtained a binary road marking segmentation in the previous chapter, but it only provides limited information in these scenarios. This chapter extends upon that work by comparing a *model-driven* and a *data-driven* approach for semantic road marking classification, which both employ the binary segmentation.

In Chapter 5, we argued that state-of-the-art DNNs have several advantages over traditional heuristic or shallow-learning pipelines for road marking segmentation and classification. They are able to leverage the global scene context [100] to distinguish

(a) **(b)**

Figure 6.1: The road markings indicating the parking space, encircled in red **(a)**, are visually and geometrically similar to the ones indicating a junction, encircled in blue **(b)**. Traditional, rule-based systems for road marking classification may experience difficulties correctly classifying the semantic type. However, DNNs leverage the scene context (i.e. parked cars or side roads), which provides enough information for accurate classification.

road marking classes with a similar geometric shape, which is difficult for rule-based systems (see Figure 6.1), and are robust to spatial deformations, degradation, and partial occlusion when adequate training data is available. However, pixel-wise ground-truth labels are often unavailable for the domain of interest, and generating these without extensive manual effort was impossible until recently.

In order to circumvent expensive manual efforts, we first demonstrate a model-driven approach that leverages domain knowledge in Section 6.1. We define a road marking as a semantic instance, which dictates particular driving behaviour given by its class and is formed by a collection of linear road marking segments, following Section 4.2.1. These linear segments are extracted directly from the binary pixels-wise segmentation by employing CORAL [101], an accurate multi-model line-fitting approach. The advantage of this approach is that it performs the optimization online and works satisfactorily in the presence of noise and an *a priori* unknown number of linear models. A second optimization step is then performed, this time jointly over the linear segments, to retrieve instances of road marking classes that are defined by their respective spatial configurations (i.e. angles and distance between the segments). We demonstrate that this approach classifies road marking instances correctly under different conditions in the reproduced publication in Appendix C.

Alternatively, road marking classification can be achieved by data-driven approaches. However, these suffer from two issues. Firstly, ground-truth labels had to be created manually for every domain of interest until recently, which

is extremely labour intensive. Secondly, simple data augmentation techniques [102] (e.g. flipping, translating, or adjusting contrast) do not deliver the necessary diversity to adapt to all encountered environments and conditions [103]. Even if more efficient hand-labelling techniques become available in the future, the issue of *edge cases*, which appear very infrequently in datasets captured during regular driving, remains. Resampling or applying a class-weighted loss function are not viable solutions for small hand-labelled datasets as these contain insufficient examples of rare classes for proper generalization. Alternatively, creating a physics-based virtual environment in which the desired road markings can be reproduced as many times as necessary is costly and requires successful domain adaptation to bridge the gap between the virtual and real world.

Fortunately, due to substantial progress in image-to-image translation [68], [104], data-driven approaches have recently become feasible. These methods allow for the reverse of the process presented in Chapter 5 (i.e. synthesizing a photo-realistic image for a predefined label). The road markings in the labels are created by following their construction guidelines. In contrast to the model-driven approach, this method is not limited by predefined linear models,and therefore extends easily to letters and symbols. This allows us to generate vast quantities of training pairs for deep road marking classification without requiring additional manual labelling effort. A new class-weighted loss function is introduced to balance the training on datasets extended with large numbers of synthetic training pairs. We show that these synthetically generated training pairs boost road marking classification for rare road markings in the reproduced publication in Section 6.2.1.

We compare the disadvantages and advantages of both approaches qualitatively in Section 6.3. The acquired semantic instances can be integrated into the scene graph to aid decision making and navigation through complex urban environments. This extension of the scene graph is not addressed in this thesis and left for further research.

In summary, this chapter makes the following principal contributions:

- A data-driven approach for road marking classification using the obtained binary segmentation to synthesize photo-realistic training images for predefined labels (T-2b). This significantly reduces the manual labelling effort (T-2c).
- A new class-weighted loss function to balance the training on datasets extended with large numbers of synthetic training pairs (T-1b).
- A real-time framework for improving the segmentation of rare road marking classes during real-world deployment in complex urban environments (T-1b).
- A comparison of the model-driven and data-driven approach to provide insight into a complete road marking classification system for real-world deployment (T-1b).

Additionally, this chapter makes the following supporting contribution in collaboration:

- A demonstration of a model-driven approach for road marking classification from the obtained binary segmentation, leveraging additional domain knowledge regarding road construction (T-1a and T-1b).

6.1 Model-Driven Road Marking Classification

This section demonstrates the model-driven approach which leverages domain knowledge for road marking classification. It operates on the premise that most painted road markings are formed by collections of simple geometric primitives (i.e. lines), even though their scale and rotation differ significantly due to the camera perspective. The spatial configuration (i.e. angles and distances) of these primitives distinguishes the road marking classes.

The entire pipeline consists of three sequential stages and takes the road marking segmentation of Chapter 5 as an input. In the first step, linear models (i.e. geometric primitives) are fitted to the segmented pixels with CORAL for several reasons. Firstly, CORAL is superior to RANSAC when the binary segmentation is not entirely accurate (i.e. contains noise). Secondly, CORAL is superior to the Hough transform

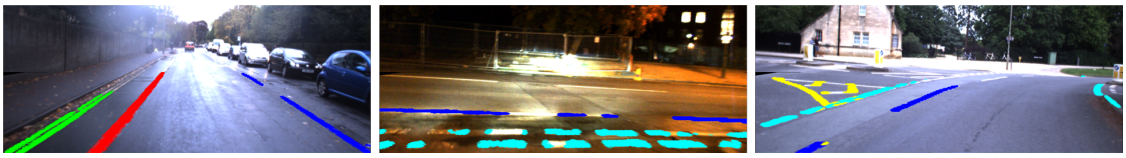


Figure 6.2: The model-driven approach employs two sequential optimization steps to retrieve road marking instances (coloured by their semantic classes) under various conditions from the binary segmentation obtained in Chapter 5.

because it depends less on correct initialization and is able to retrieve the optimal number of models without knowing this *a priori*. Lastly, CORAL is parallelisable and therefore runs online. In the second step, a subsequent optimization step clusters these linear segments into semantically meaningful classes by two types of constraints: the angles and distances between the linear segments according to the definitions for road construction. In the final step, road markings are tracked from frame to frame to improve temporal robustness and solve ambiguities in case of occlusions.

A reproduction of the publication with more details is provided in Appendix C. Therein, it was demonstrated that this approach is able to classify six selected types of road markings under different weather conditions, as visualised in Figure 6.2. The presented approach is compared to the data-driven alternative in Section 6.3.

6.2 Data-Driven Road Marking Classification

This section presents our data-driven approach for boosting road marking classification without requiring additional labelling effort. Training pairs are generated by altering semantic maps of real-world scenes and subsequently synthesizing a corresponding photo-realistic image with a GAN.

We place instances of desired road markings randomly onto a blank road surface to avoid the difficult problem of composing natural scenes, which has become more feasible after the publication of our work [105]. Our approach is inspired by the principles of domain randomization [106], and we, therefore, refer to it as *road layout randomization*. We specifically target road marking classes that appear infrequently in datasets collected during regular driving and demonstrate that the synthetic training pairs improve generalization during real-world deployment.

Furthermore, we introduce a new class-weighted cost function, which ensures that the performance for other road marking classes is retained in the presence of vast quantities of synthetic training pairs of a rare class. This approach is presented in full detail in the reproduced publication in Section 6.2.1. Moreover, we provide further details regarding the newly-introduced cost function, additional qualitative results, and a discussion regarding the quality of the generated synthetic images. The presented approach is compared to the model-driven alternative in Section 6.3.

6.2.1 Publication

This section contains a reproduction of the following publication:

> [20] **T. Bruls**, H. Porav, L. Kunze, and P. Newman, "Generating all the roads to Rome: Road layout randomization for improved road marking segmentation", in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Oct. 2019, pp. 831-838.

Generating All the Roads to Rome: Road Layout Randomization for Improved Road Marking Segmentation

Tom Bruls, Horia Porav, Lars Kunze, and Paul Newman

***Abstract*— Road markings provide guidance to traffic participants and enforce safe driving behaviour, understanding their semantic meaning is therefore paramount in (automated) driving. However, producing the vast quantities of road marking labels required for training state-of-the-art deep networks is costly, time-consuming, and simply infeasible for every domain and condition. In addition, training data retrieved from virtual worlds often lack the richness and complexity of the real world and consequently cannot be used directly. In this paper, we provide an alternative approach in which new road marking training pairs are automatically generated. To this end, we apply principles of domain randomization to the road layout and synthesize new images from altered semantic labels. We demonstrate that training on these synthetic pairs improves mIoU of the segmentation of rare road marking classes during real-world deployment in complex urban environments by more than 12 percentage points, while performance for other classes is retained. This framework can easily be scaled to all domains and conditions to generate large-scale road marking datasets, while avoiding manual labelling effort.**

I. Introduction

Safety-critical systems, such as automated vehicles, need interpretable and explainable decision-making for real-world deployment. An important aspect for improving interpretability of such systems is the ability to explain scenes semantically. More specifically, planning the behaviour of an automated vehicle through an urban traffic environment requires understanding of the *road rules*. These are conveyed to the traffic participants by the markings painted on the road.

Although semantic reasoning about road markings is ideally performed at an object and scene level [1], state-of-the-art deep learning methods perform semantic segmentation at the pixel level. This, however, requires thousands of pixel-labelled images for different environments and conditions, which is a problem for several reasons. Firstly, it is impossible to label every pixel of every image for every city in every condition manually. Secondly, simple data augmentation techniques [2] (e.g. flipping, translating, adjusting contrast, etc.) do not deliver the necessary diversity to adapt to all encountered environments and conditions [3].

Even if more efficient hand-labelling techniques become available in the future, we still face the issue of *edge cases* that appear very infrequently in regular driving. In the context of road marking segmentation, data collection during regular driving creates extremely imbalanced datasets. For example, zigzag markings (which indicate a pedestrian crossing, Fig. 1) are encountered rarely, but their detection is critical for safe operation. Resampling or applying a class-weighted loss function are not viable solutions for small, hand-labelled datasets, since these simply contain insufficient examples of rare classes for proper generalization. Retrieving more examples is labour intensive in terms of driving and labelling time. Consequently, trained classifiers show decreased performance on infrequently-occurring classes [4].

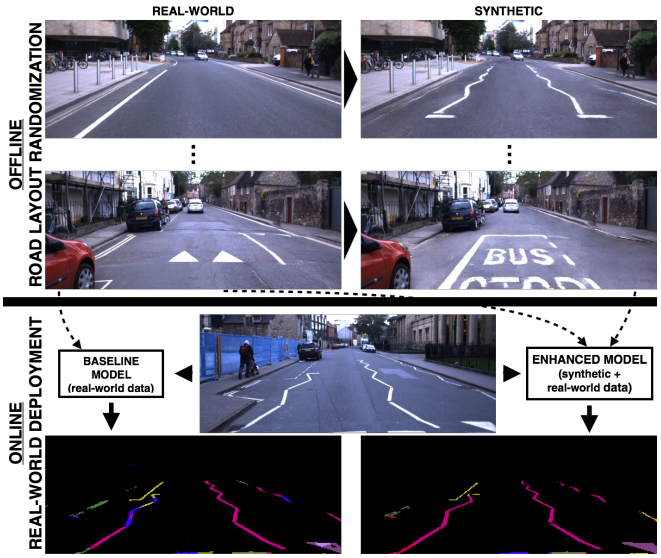


Fig. 1. Road layout randomization for improved road marking segmentation, while avoiding manual labelling. *Offline:* Firstly, new images for training road marking segmentation networks are automatically generated by synthesizing new road surfaces from altered semantic labels. *Online:* Subsequently, mIoU of the segmentation of rare road markings (e.g. zigzags shown in pink) is improved by more than 12 percentage points during *real-world* deployment by the enhanced model trained on a hybrid dataset when compared to a baseline model only trained on the real-world dataset.

The latter problem could be solved by creating a virtual environment (i.e. simulator), in which the desired road markings can be reproduced as many times as necessary. However, this introduces several new challenges. Firstly, even though state-of-the-art simulators can appear realistic to the human-eye, their fidelity lacks the richness and complexity of the real world and consequently there is still an apparent domain gap between simulated environments and their real-world equivalent. As a result, domain adaptation techniques need to be applied for real-world deployment [5], [6]. Secondly, although we might be able to generate simulated environments from real-world data in the future [7], at present their design remains a manual, costly, and time-consuming task. Besides, since urban environments can vary substantially between countries, there is a need for highly-configurable

Authors are from the Oxford Robotics Institute, Dept. Engineering Science, University of Oxford, UK. {tombruls, horia, lars, pnewman}@robots.ox.ac.uk

virtual worlds, which increases the labour cost.

Recently, alternative methods have been developed [8] to synthesize new, photo-realistic scenes for a domain of interest by employing Generative Adversarial Networks (GANs). These approaches require relatively little human effort and can easily extend to all kinds of different conditions [9]. This provides the ability to generate large-scale datasets for semantic scene understanding in a domain of interest at low cost. Most of these frameworks take real-world scenes and augment them by placing or removing objects (e.g. cars, pedestrians, etc.). This can be done randomly [10] or more naturally by learning from real-world examples [11], [12].

Similarly, we place instances of chosen road markings into newly-synthesized, photo-realistic scenes, which are then used to train a road marking segmentation network. In this way, we generate sufficient examples of *rare* road marking classes to achieve the generalization performance required during real-world deployment, as visualized in Fig. 1. However, placing new road markings coherently into the scene is difficult, since there are many dependencies such as the type of road / intersection, traffic lights, parked cars, etc. that need to be taken into account. We avoid solving this hard problem by employing the principles of domain randomization [13]. More concretely, we place road markings at random places on the road surface, not necessarily coherent with other elements in the scene. In this way, we perform *road layout randomization*. Real-world scenes encountered during deployment then appear as samples of the broadened distribution on which the model was trained.

We demonstrate quantitatively that training on these synthetic labels improves mIoU of the segmentation of rare road marking classes, for which it is expensive to attain sufficient real-world examples, during real-world deployment in complex urban environments by more than 12 percentage points. To take full advantage of the synthetic labels we introduce a new class-weighted cross-entropy loss which balances the training. Furthermore, we show qualitatively that the segmentation performance for other classes is retained.

We make the following contributions in this paper:

- We present a method for generating large-scale road marking datasets for a domain of interest by leveraging principles of domain randomization, while avoiding expensive manual effort.
- We introduce a new class-weighted cross-entropy loss to balance the training on synthetic datasets with large class-wise imbalance in terms of their occurrence.
- We demonstrate a real-time framework for improving the segmentation of (rare) road marking classes in *real-world*, complex urban environments.

II. Related Work

Road Marking Segmentation: Deep networks are increasingly used to perform lane detection in highway scenarios [14]–[16]. However, the urban environments and road markings targeted in this paper are substantially different and more complex, and thus require a different approach. This problem has seen significantly fewer deep learning solutions, due to a lack of large-scale datasets containing road markings. The first large-scale semantic road marking dataset was recently introduced in [17], however it is extremely expensive to manually expand this to all environments and conditions.

Road marking segmentation as demonstrated in [4] is closest to the application of this paper. The authors train a network for semantic road marking segmentation and improve their results by predicting the vanishing point simultaneously. In contrast to this paper, they require thousands of hand-labelled images, which is very labour expensive. Alternatively, the authors of [18] hand-label road markings such as arrows and bicycle signs and train an object detection network to predict bounding boxes instead of pixel segmentations. In previous work [19] (includes more extensive review), we have introduced a weakly-supervised approach for binary road marking segmentation, which is used here to acquire road marking labels for real-world scenes.

Synthetic Training for Automated Driving Tasks: To prevent costly and time-consuming manual labelling of training data, many approaches leverage synthetic datasets. Early works trained on purely virtual data to perform object detection [20], [21] or semantic segmentation [5], [6].

However, virtual data lacks the richness and complexity of the real world. A possible alternative is to augment real-world data. For the task of semantic segmentation this means either generating new, photo-realistic images from semantic labels [8], [22], [23] or enriching semantic labels with virtually-generated information [24]. Both of these principles are applied in this paper. For object detection tasks, the main difficulty is to place the (dynamic) objects coherently into the scene. The simplest solution is random object placement (i.e domain randomization) [10]. Alternatively, the authors of [25], [26] place photo-realistic, synthetic cars into real-world images by taking into account the geometry of the scene. The most recent approaches [11], [12], [27], [28] learn context-aware object placement from real-world examples. However, placing dynamic objects, such as pedestrians, seems less complex than road markings, because the space of realistic solutions is less restrictive. Therefore, we place road markings randomly onto the road surface in this paper.

Scene Manipulation: Recently, several approaches have been introduced for more complex scene manipulation, beyond simple augmentation. Additional sensor modalities are used in [29] to offer the flexibility (e.g. different view points) of a virtual simulator, while generating data with the fidelity and richness of real-world images. The authors of [30] introduce a probabilistic programming language to synthesize complex scenarios from existing domain knowledge. Another system [31] offers similar levels of control, while the camera sensor is modelled accurately at the same time. These frameworks potentially offer a way to generate improved training data for our approach.

III. Generating Synthetic Training Pairs

In this section, we explain in detail how to generate synthetic training pairs for road marking segmentation networks to improve performance during real-time deployment, as

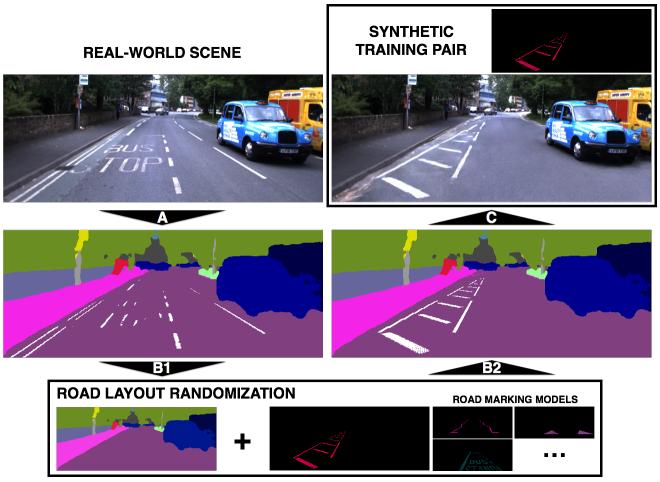


Fig. 2. Road layout randomization: generating synthetic training data based on real-world scenes. The process has the following steps (as described in the respective subsections of Section III): (A) semantic segmentation of the real-word scene is acquired, (B1) the road markings are removed and replaced with road surface, (B2) instances of chosen road markings (modelled according to the UK Highway Code) are placed randomly on the road surface, and finally (C) the road surface of the original image is replaced with a GAN-synthesized, photo-realistic alternative based on the altered semantic label. The composite image is then paired with the generated road marking label.

shown in Fig. 2. We demonstrate that this framework can be employed on any driving dataset even when no ground-truth semantic or road marking labels are available.

A. Retrieving Semantic Labels for Real-World Scenes

In order to generate synthetic training pairs for road marking segmentation, the road layout of semantic labels of real-world scenes is altered and from these new, photo-realistic images are synthesized. Ground-truth semantic labels are not required for the domain of interest, since semantic segmentation of reasonable (i.e. sufficient) quality can be acquired from a model pretrained on the Cityscapes dataset[1]. In this way, we retrieve semantic labels of real-world scenes from the Oxford RobotCar dataset [32], as shown in Fig. 3.

Unfortunately, the available model is not trained to segment road markings (Cityscapes does not contain road marking masks). However, semantic labels including road markings and their corresponding real-world images are necessary to train the GAN described in Section III-C. We prevent manual labelling of road markings by employing the techniques of [19] to generate large quantities of road marking annotations automatically. Because these annotations are generated automatically, they are not equivalent to the ground-truth, however they have proven to be sufficient for training purposes if regularization techniques are applied. The road markings are added to the semantic labels acquired from the Cityscapes model, as visualized in Fig. 3.

B. Road Layout Randomization

To form new road marking training pairs, we alter the road layout (i.e. road markings) of the retrieved semantic labels

[1]https://github.com/tensorflow/models/blob/master/research/deeplab/g3doc/model_zoo.md

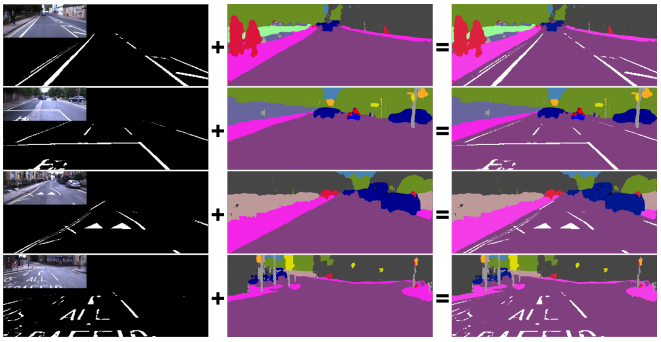


Fig. 3. We augmented the Oxford Robotcar Dataset with semantic labels including road markings to train the CGAN described in Section III-C. The semantic segmentation label is retrieved from inference with a pretrained Cityscapes model and combined with automatically generated road marking annotations from [19]. The resulting labels are not perfect ground-truth, but they are sufficient for the task and can be acquired at low cost.

and subsequently synthesize a new corresponding image. In order to rebalance datasets collected during regular driving, we create new semantic labels with road markings which occur relatively infrequently in the real world (e.g. pedestrian crossing, arrows, etc.). By training the road marking segmentation network on the rebalanced dataset, the goal is to improve the performance for these respective *rare* classes, while at the same time retaining the overall performance.

As mentioned before, the type and placement of road markings is dependent on many factors of the scene such as the type of road, traffic lights, and even the traffic participants. Altering all of these coherently according to the real world is difficult and seems similar in terms of complexity to designing a simulator. Therefore, we choose to leverage domain randomization principles [10]. We vary position (and scale accordingly), rotation, quantity, and partial occlusion of the road markings that are placed into the environment and in that way perform *road layout randomization* to create vast quantities of new training pairs automatically. For accurate placement, we use the camera sensor calibration of the vehicle and assume that the road surface is planar and horizontal. Training the network on many randomly-generated pairs improves generalization in newly-encountered, real-world scenes, which then appear as variations of the distribution on which the network was trained.

Concretely, we start by erasing the original road markings from the real-world semantic labels and subsequently place a new road marking instance onto the cleared road surface. The classes are realistically modelled according to the UK Highway Code so that their shape, size, colour and configuration (e.g. zigzags appear in dual or triple configurations) resemble the real world. Some examples for different classes of rare road markings are given in Fig. 4.

C. Synthesizing Photo-Realistic Images

In order to create a synthetic training pair, we train a Conditional Generative Adversarial Network (CGAN), as introduced in [8], to synthesize a photo-realistic RGB image for the altered semantic label (from Section III-B). In this

Fig. 4. Examples of newly-synthesized training images for several rare road marking classes (i.e. zigzag, diagonal stripes, bus stop, and small warning triangles) by employing road layout randomization. The top two rows show images of real-world scenes of the Oxford RobotCar dataset together with the corresponding (partial) semantic labels (Section III-A). The third row visualizes the altered semantic labels in which instances of chosen road markings are placed randomly on the road surface (Section III-B). The last row presents the newly-synthesized road surfaces substituted into the real-world images. The GAN is able to generate road surfaces with photo-realistic textures, lighting, and even degradation as exemplified on the letters of the bus stops.

framework the generator G aims to synthesize the RGB images, while the discriminator D tries to distinguish synthesized from real-world images. The CGAN is trained in a supervised setting using real-world images and corresponding semantic labels retrieved in Section III-A. After the training is completed a photo-realistic image can be synthesized by the generator from the altered semantic labels generated in Section III-B, as shown in Fig. 4.

More specifically, the framework incorporates several advancements over previous works which make it possible to generate higher-resolution images. Firstly, the generator architecture follows a traditional downsample-bottleneck-upsample model, but splits into a global generator and a local enhancer, where the local component is forced to learn high-resolution details for the stabilized features of the global component. Secondly, to overcome discriminator capacity limitations which arise from training with high-resolution images, the framework incorporates three similar discriminators that work on different scales. The discriminators with bigger receptive field enforce more globally consistent image generation, while the smaller receptive fields steer the generator towards more realistic, fine-level details. Lastly, the traditional GAN loss is augmented to include a feature matching loss based on the discriminator. Formally, following the architecture described in [8], given $K = 3$ discriminators D_k, each operating on a different scale, along with the input and label images $I^{\text{SEG}}_{\text{input}}$ and $I^{\text{RGB}}_{\text{label}}$, respectively, the final objective to be minimized is:

$$\mathcal{L}_{\text{tot}} = \min_G ((\max_{D_1,D_2,D_3} \sum_{k=1,2,3} \mathcal{L}_{\text{GAN}}(G, D_k)) + \lambda_{\text{FM}} \sum_{k=1,2,3} \mathcal{L}_{\text{FM}}(G, D_k) + \lambda_{\text{VGG}} \mathcal{L}_{\text{VGG}}(G)). \tag{1}$$

Here, $\mathcal{L}_{\text{GAN}}(G, D_k)$ represents the usual GAN loss (see [8]) defined over K scales, $\mathcal{L}_{\text{FM}}(G, D_k)$ is the discriminator feature loss defined over K scales:

$$\mathcal{L}_{\text{FM}(G,D_k)} = \sum_{i=1}^{l_D} \frac{1}{w_i} \| D_k(I^{\text{RGB}}_{\text{label}})_i - D_k(G(I^{\text{SEG}}_{\text{input}}))_i \|_1, \tag{2}$$

with l_D defining the number of layers from the discriminator used in the discriminator feature loss and $\mathcal{L}_{\text{VGG}}(G)$ being the perceptual loss:

$$\mathcal{L}_{\text{VGG}(G)} = \sum_{i=1}^{l_P} \frac{1}{w_i} \| \text{VGG}(I^{\text{RGB}}_{\text{label}})_i - \text{VGG}(G(I^{\text{SEG}}_{\text{input}}))_i \|_1, \tag{3}$$

with l_P defining the number of layers from an ImageNet-trained network (in this case VGG16) used in computing the perceptual loss. The factors $w_i = 2^{l-i}$ are utilized to scale the weight of each network layer used in computing the losses. We train the model on 3351 overcast training pairs while using the settings as specified in [8] to generate images with a resolution of 256×640.

Unfortunately, the RobotCar dataset does not contain any boundary or instance labels (as used in [8]) necessary to generate sharp, high-quality images. Consequently, the generated images can be smudgy around object boundaries (e.g. rows of parked cars are merged because of the image perspective, as exemplified in [8]) and contain unnatural artifacts. Therefore, we choose to substitute only the newly-generated road surface and keep the rest of the original image intact. The RobotCar dataset contains sufficient real-world images so that no background duplicates have to exist in the new road marking dataset. In this way, we are able to generate a large-scale urban datasets for road marking segmentation, while avoiding expensive manual labelling.

The above-described framework can easily be extended to different (weather and lighting) conditions by training condition-specific models. If it is not possible to retrieve semantic labels of sufficient quality under difficult conditions, a state-of-the-art invertible generator, that can transform the images into the desired appearance similar to [9], [33], can be employed. In this way the semantic label acquired from the

overcast image can be paired with an image which resembles a different weather or lighting condition.

IV. Training for Road Marking segmentation

In this section, the network trained for road marking segmentation is described in detail, along with some important considerations that have to be taken into account when rebalancing datasets.

A. Network Architecture

Deep networks for road marking segmentation have several advantages over traditional heuristic or shallow-learning pipelines. Firstly, they are more robust to spatial deformations, degradation, and partial occlusion. Secondly, the scene context can be leveraged to improve semantic segmentation and thereby understand the road rules. For instance, similarly-shaped road markings (e.g. lane separators and separators that mark a parking spot) can be classified differently based on their place in the scene and relationship with other objects, whereas this is difficult to accomplish with traditional rule-based systems.

We train a U-Net model [34], but include batch normalization and dropout as regularization techniques. These are paramount in our framework, since we train on partial labels that are generated automatically. Dropout allows the network to extend its prediction towards road marking pixels that were wrongly assigned to the background in the partial labels, because they share more similarities with the road marking class than the background class. The architecture and training settings used are similar to our previous work [19], with the major exception that the output now predicts multiple classes of road markings instead of a binary segmentation. More specifically, the output of the network is computed by applying a channel-wise softmax activation over the final feature maps and assigning a class to each respective pixel by taking the channel-wise $\arg\max$ over the output channels, yielding a one-channel discrete class activation map.

At run time, the Tensorflow implementation of the network performs inference on an input image in real-time ($\sim$62.5 Hz) on an NVIDIA TITAN Xp GPU.

B. Balancing of the Classes

As mentioned before, datasets collected during regular driving are extremely imbalanced in terms of the occurrences of particular road marking classes. For instance, zigzag markings are only found in $\sim 7\%$ of the images, whereas lane separators occur in $\sim 70\%$. Solutions such as resampling the dataset or applying a class-weighted loss function are not viable for small, hand-labelled datasets, because they simply contain an insufficient number of examples of the rare classes to generalize well to unseen cases during deployment.

In this paper, we opt for a different approach in which we synthesize new training pairs for rare classes automatically and add them to an existing dataset. This ensures that there are enough examples of these classes for the network to learn from. However, it is not obvious how to produce a rebalanced dataset including synthetic training pairs that is optimal for training. To counteract the fact that we might add too many synthesized training pairs, we experiment with three types of class-weighted cross-entropy losses:

1) Equal weighting (EQ) of all classes irrespective of their occurrence in the dataset.
2) Median frequency balancing (FB) [35], in which each pixel is weighted by
$$w_c = \frac{\text{median}\,(F)}{f_c}, \tag{4}$$
where $F = \{f_1, \ldots, f_C\}$ with f_c denoting the total number of pixels of class c divided by the total number of pixels in labels where c is present and C the total number of classes.
3) Median total balancing (TB), in which each pixel is weighted by
$$w_c = \frac{\text{median}\,(G)}{f_c + n_c}, \tag{5}$$
where $G = \{f_1 + n_1, \ldots, f_C + n_C\}$ with f_c equivalent to 2) and n_c denoting the number of labels in which class c is present divided by the total number of training pairs.

It is important to note that median frequency balancing only corrects for the fact that some classes naturally occupy less pixels in the images. For instance, dotted lines indicating a pedestrian crossing are smaller in accumulated area than an alternative zebra crossing. However, median frequency balancing does not account for imbalance in occurrences across the dataset; whether $\sim 7\%$ of the images contain zigzag markings or $\sim 70\%$, the weight remains the same as long as their pixel size remains equivalent. This is not ideal, since we artificially create an imbalance in the number of occurrences by adding labels of specific classes. The third weighting function, introduced in this paper, is designed to take this into account, balancing the average pixel area as well as the imbalance in occurrences across the dataset.

V. Experimental Results

In this section we describe the experimental setup and the datasets that we have created, before we present the quantitative and qualitative results.

A. Experimental Setup

We have selected four types of rare road markings for evaluation: bus stops, diagonal stripes (must not enter), small warning triangles, and zigzag markings. These classes function as a proof of concept, but the framework can be applied to any class (i.e. model) of road markings. For quantitative pixel-wise evaluation, we have hand-labelled 102, 102, 96, and 102 *real-world* images containing bus stops, diagonal structures, small warning triangles, and zigzag markings, respectively. Note that in these images only these respective classes were labelled and all other classes present were ignored (see Fig. 7). While we train all models to predict the *full* set of 20 different road markings and show these results qualitatively, we only evaluate the four selected classes quantitatively. We define the pixel-wise metrics $\text{PRE} = \frac{\text{TP}}{\text{TP+FP}}$, $\text{REC} = \frac{\text{TP}}{\text{TP+FN}}$, $\text{F}_1 = 2 * \frac{\text{PRE*REC}}{\text{PRE+REC}}$, and

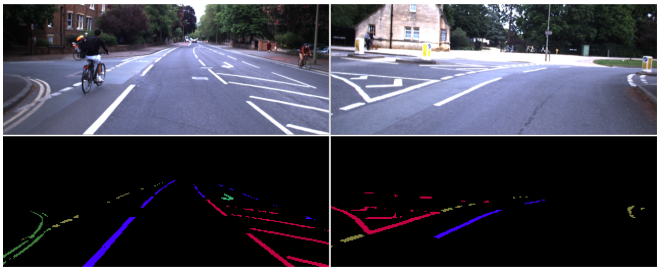


Fig. 5. Examples of the partial labels created by semantically classifying the binary annotations of [19]. Although not perfect ground-truth, these labels can be used to train a baseline model to predict the full set of road markings.

$\text{IoU} = \frac{\text{TP}}{\text{TP+FP+FN}}$ with TP, FP, and FN denoting the true positive, false positive, and false negative pixels, respectively. In contrast to binary classification, all metrics are evaluated at the operating point defined by taking the channel-wise $\arg\max$ over the multi-class output on a per image basis and averaged over the test set, without any further fine tuning of the operating characteristics. Furthermore, we have hand-labelled 25 real-world images for each respective class for validation. We train until convergence and select the epoch for testing in which the mIoU is highest among the evaluations on the validation set. It should be noted that road marking segmentation is arguably a harder task than scene segmentation, because road marking elements are fairly small in general, often degraded, and the different types share many visual and geometric similarities. State-of-the-art approaches achieve a mIoU of around 40%, however a benchmark has only been established recently [17].

As a reasonable baseline, 1000 partial, binary labels generated by [19] collected during regular driving were hand-labelled class-wise. Although not equivalent to the ground-truth, we have proven in [19] and will demonstrate again in Section V-C that these labels are sufficient to achieve full segmentation, when regularization techniques are applied. A few examples are given in Fig. 5. The labels contain the 20 different types of road markings, so that the network functions as a full road marking segmentation system. However, many classes occur too infrequently to achieve state-of-the-art performance, because the network fails to generalize to new scenarios during deployment. For instance, the baseline dataset only contains 63, 109, 39, and 74 images with bus stops, diagonal stripes, small warning triangles, and zigzag markings, respectively. For the other experiments, we add synthetic training pairs of the four classes to the baseline dataset. In this way, the network still predicts all 20 classes, but is given a sufficient number of labels of the rare classes to improve generalization during real-world deployment.

B. Quantitative Evaluation

In order to understand how the number of added synthetic images influences the performance, we have added different numbers of synthetic zigzag pairs to the baseline dataset, while keeping the other classes constant. The results for the three different cross-entropy losses are presented in Fig. 6.

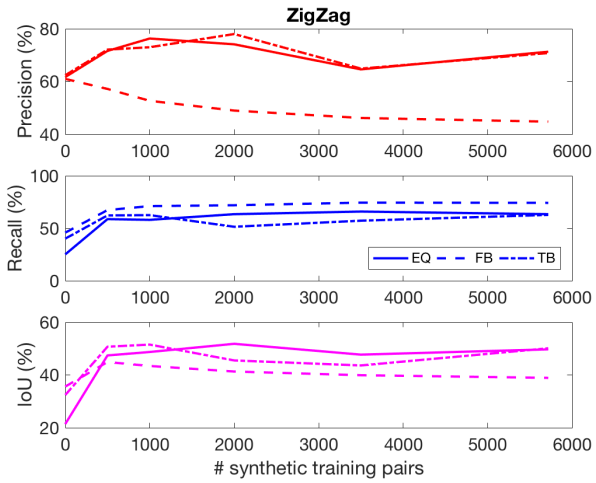


Fig. 6. Pixel-wise performance of zigzag segmentation when training with different cross-entropy losses for a variable number of synthetic images added to the baseline dataset.

The following key observations can be made:

- Adding as little as 500 synthetic training pairs already makes a substantial difference in terms of overall performance.
- Adding more than 2000 synthetic training pairs does not provide extra benefits in general. Further performance increase beyond this level might require higher-quality, more diverse, more coherent synthetic images.
- FB struggles to balance training as more synthetic pairs are added, due to the fact that it does not account for occurrence imbalance across the dataset. The precision drops significantly as the network learns from an abundance of zigzag markings and starts classifying other classes incorrectly as zigzag.
- TB alleviates the precision drop of FB, but does not outperform EQ consistently among all metrics.

Assuming that these observations hold similarly for the other classes, 1000 synthetic training pairs of each respective class were added to the baseline dataset as a proof of concept to train enhanced networks (with the different loss functions). From the results, as presented in Table I, the following key observations can be made:

- By adding synthetic training pairs, IoU performance similar to the state-of-the-art can be achieved when only very few real-world examples are available. mIoU is increased by 12.4% (comparing the best baseline and enhanced models) without using any manual labelling effort.
- The enhanced networks always achieve better overall performance (i.e IoU) by a substantial margin for the equivalent cost function. Segmentation performance can thus be boosted cheaply by the presented framework.
- TB outperforms FB in terms of F_1 and IoU in general, because it accounts for the class imbalance across the dataset that was artificially created by adding synthetic pairs. TB offers a good trade-off between high precision achieved by EQ and high recall achieved by FB.

TABLE I

PIXEL-WISE PERFORMANCE FOR RARE CLASSES FOR THE BASELINE (B) AND ENHANCED (E) MODELS

		BUS STOP				DIAGONAL				TRIANGLE				ZIGZAG				MEAN			
Model	**Loss**	**PRE**	**REC**	**F_1**	**IoU**	**PRE**	**REC**	**F_1**	**IoU**	**PRE**	**REC**	**F_1**	**IoU**	**PRE**	**REC**	**F_1**	**IoU**	**PRE**	**REC**	**F_1**	**mIoU**
B	EQ	61.6	17.8	27.6	16.1	59.1	24.6	34.7	21.8	60.7	41.1	49.0	34.4	65.9	22.5	33.6	20.1	61.8	26.5	36.2	23.1
B	FB	64.7	26.3	37.3	22.8	58.8	31.0	40.6	26.4	60.1	47.7	53.2	36.9	64.7	34.8	45.3	29.5	62.1	35.0	44.1	28.9
B	TB	62.2	19.1	29.2	17.1	58.4	33.3	42.4	27.7	59.9	51.6	53.4	39.4	62.3	31.3	41.7	26.5	60.7	33.8	42.2	27.7
E	EQ	**74.8**	28.5	41.2	26.3	**73.0**	40.9	52.4	35.8	**61.4**	46.9	53.2	38.4	**69.4**	49.8	58.0	40.7	**69.7**	41.5	51.2	35.3
E	FB	54.8	**62.9**	**58.6**	**40.1**	45.8	**67.4**	54.6	39.1	46.9	73.8	57.3	37.9	51.0	**66.6**	57.8	40.3	49.6	**67.7**	57.1	39.4
E	TB	58.5	55.3	56.8	39.2	51.4	59.1	**55.0**	**40.2**	50.8	**75.1**	**60.6**	**43.4**	61.4	57.3	**59.3**	**42.5**	55.5	61.7	**57.9**	**41.3**

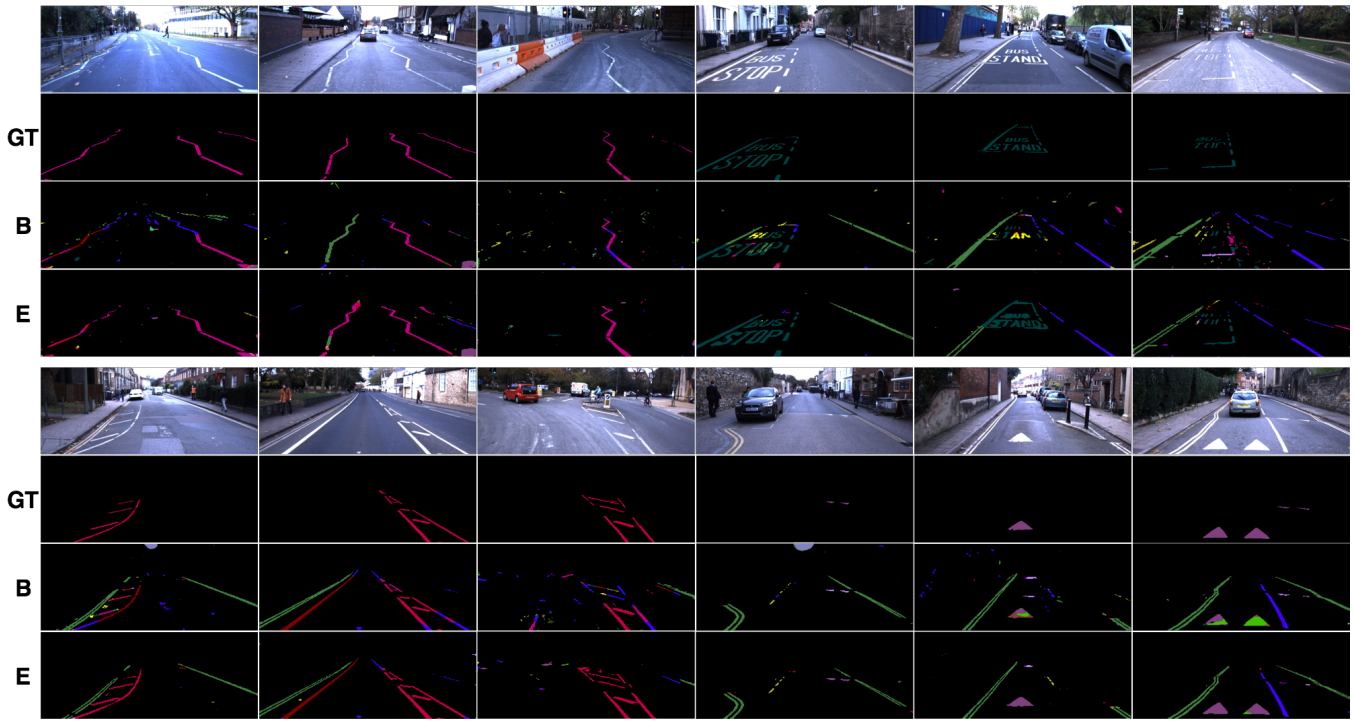


Fig. 7. Road marking segmentation (full set of classes) in traffic environments with rare classes. The *top two* rows of each scene show the input image together with the corresponding ground-truth (GT) label of the rare class, which is used for quantitative evaluation. The *bottom two* rows of each scene depict the segmentation results for the best performing baseline (B) and enhanced model (E), respectively. The enhanced model provides more consistent and correct segmentation of the rare classes, while retaining reasonable and sometimes achieving improved performance for other classes (e.g. *green* double boundaries, *blue* separators, *yellow* parking spot separators, etc.).

C. Qualitative Evaluation

In Fig. 7, the best baseline and enhanced models are compared qualitatively for different traffic scenes. All networks are trained to predict the full set of 20 different road marking classes, however scenes with the respective rare classes are selected for visualization.

It is clear that adding synthetic images to the training set results in more consistent and correct segmentation of the rare classes, while retaining reasonable and sometimes achieving improved performance for other classes. The latter could be caused by the general increase of the number of training examples and/or better balancing of the cost function. The enhanced model trained with TB offers more satisfying (i.e. less noisy) visual results than the baseline model trained with FB. Furthermore, it is clear that full segmentation of the road marking elements is possible from partial labels when regularization techniques are applied correctly. Thus, this framework offers an effective and efficient step towards a road marking classification system for automated driving pipelines.

VI. CONCLUSION

We have presented a weakly-supervised approach for improving road marking segmentation in complex urban environments. To this end, we alter semantic labels of real-world scenes with instances of chosen road markings using domain randomization principles and synthesized corresponding, photo-realistic images to generate vast quantities of synthetic training pairs, thereby avoiding the need for expensive manual labelling. During deployment, we predict 20 classes of road markings in real time and we have demonstrated quantitatively that this framework improves mIoU of rare classes by more than 12 percentage points and thus reaches state-of-the-art performance with very few real-world labels. This is achieved by introducing a new class-weighted cross-entropy loss to balance the training of

synthetic datasets. Furthermore, we have shown qualitatively that the segmentation performance for other classes is retained. The presented framework can easily be extended to include other classes or work under different conditions and results can be expected to improve as more advanced synthesizing networks will emerge in the future. Hence, road layout randomization is an effective and efficient technique to enhance road marking classification systems in automated driving pipelines.

ACKNOWLEDGMENT

The work has been supported by the EPSRC/UK Research and Innovation Programme Grant EP/M019918/1 (Mobile Autonomy: Enabling a Pervasive Technology of the Future). We acknowledge the support of NVIDIA Corporation with the donation of Titan Xp and Titan V GPUs.

REFERENCES

[1] L. Kunze, T. Bruls, T. Suleymanov, and P. Newman, "Reading between the lanes: Road layout reconstruction from partially segmented scenes," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, Nov 2018, pp. 401–408.

[2] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, "Autoaugment: Learning augmentation policies from data," *arXiv preprint arXiv:1805.09501*, 2018.

[3] R. Krajewski, T. Moers, and L. Eckstein, "VeGAN: Using GANs for augmentation in latent space to improve the semantic segmentation of vehicles in images from an aerial perspective," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Jan 2019, pp. 1440–1448.

[4] S. Lee, J. Kim, J. S. Yoon, S. Shin, O. Bailo, N. Kim, T. Lee, H. S. Hong, S. Han, and I. S. Kweon, "VPGNet: Vanishing point guided network for lane and road marking detection and recognition," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017, pp. 1965–1973.

[5] Y. Chen, W. Li, X. Chen, and L. Van Gool, "Learning semantic segmentation from synthetic data: A geometrically guided input-output adaptation approach," *arXiv preprint arXiv:1812.05040*, 2018.

[6] A. Dundar, M.-Y. Liu, T.-C. Wang, J. Zedlewski, and J. Kautz, "Domain stylization: A strong, simple baseline for synthetic to real image domain adaptation," *arXiv preprint arXiv:1807.09384*, 2018.

[7] R. Cura, J. Perret, and N. Paparoditis, "Streetgen: In base city scale procedural generation of streets: road network, road surface and street objects," *arXiv preprint arXiv:1801.05741*, 2018.

[8] T. Wang, M. Liu, J. Zhu, A. Tao, J. Kautz, and B. Catanzaro, "High-resolution image synthesis and semantic manipulation with conditional GANs," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2018, pp. 8798–8807.

[9] T. Park, M.-Y. Liu, T.-C. Wang, and J.-Y. Zhu, "Semantic image synthesis with spatially-adaptive normalization," *arXiv preprint arXiv:1903.07291*, 2019.

[10] J. Tremblay, A. Prakash, D. Acuna, M. Brophy, V. Jampani, C. Anil, T. To, E. Cameracci, S. Boochoon, and S. Birchfield, "Training deep networks with synthetic data: Bridging the reality gap by domain randomization," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, June 2018, pp. 1082–10 828.

[11] D. Lee, S. Liu, J. Gu, M.-Y. Liu, M.-H. Yang, and J. Kautz, "Context-aware synthesis and placement of object instances," in *Advances in Neural Information Processing Systems*, 2018, pp. 10 414–10 424.

[12] A. Prakash, S. Boochoon, M. Brophy, D. Acuna, E. Cameracci, G. State, O. Shapira, and S. Birchfield, "Structured domain randomization: Bridging the reality gap by context-aware synthetic data," *arXiv preprint arXiv:1810.10093*, 2018.

[13] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2017, pp. 23–30.

[14] B. De Brabandere, W. Van Gansbeke, D. Neven, M. Proesmans, and L. Van Gool, "End-to-end lane detection through differentiable least-squares fitting," *arXiv preprint arXiv:1902.00293*, 2019.

[15] N. Garnett, R. Cohen, T. Pe'er, R. Lahav, and D. Levi, "3D-LaneNet: end-to-end 3D multiple lane detection," *arXiv preprint arXiv:1811.10203*, 2018.

[16] M. Ghafoorian, C. Nugteren, N. Baka, O. Booij, and M. Hofmann, "EL-GAN: Embedding loss driven generative adversarial networks for lane detection," in *Computer Vision – ECCV 2018 Workshops*, L. Leal-Taixé and S. Roth, Eds. Cham: Springer International Publishing, 2019, pp. 256–272.

[17] X. Huang, X. Cheng, Q. Geng, B. Cao, D. Zhou, P. Wang, Y. Lin, and R. Yang, "The ApolloScape dataset for autonomous driving," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, June 2018, pp. 1067–10 676.

[18] T. M. Hoang, P. H. Nguyen, N. Q. Truong, Y. W. Lee, and K. R. Park, "Deep retinanet-based detection and classification of road markings by visible light camera sensors," *Sensors*, vol. 19, no. 2, 2019.

[19] T. Bruls, W. Maddern, A. A. Morye, and P. Newman, "Mark yourself: Road marking segmentation via weakly-supervised annotations from multimodal data," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 1863–1870.

[20] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig, "Virtualworlds as proxy for multi-object tracking analysis," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 4340–4349.

[21] M. Johnson-Roberson, C. Barto, R. Mehta, S. N. Sridhar, K. Rosaen, and R. Vasudevan, "Driving in the matrix: Can virtual worlds replace human-generated annotations for real world tasks?" in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, May 2017, pp. 746–753.

[22] S. Liu, J. Zhang, Y. Chen, Y. Liu, Z. Qin, and T. Wan, "Pixel level data augmentation for semantic image segmentation using generative adversarial networks," *arXiv preprint arXiv:1811.00174*, 2018.

[23] K. Li, T. Zhang, and J. Malik, "Diverse image synthesis from semantic layouts via conditional IMLE," *arXiv preprint arXiv:1811.12373*, 2018.

[24] Q. Geng, F. Lu, X. Huang, S. Wang, X. Cheng, Z. Zhou, and R. Yang, "Part-level car parsing and reconstruction from single street view," *arXiv preprint arXiv:1811.10837*, 2018.

[25] H. A. Alhaija, S. K. Mustikovela, L. Mescheder, A. Geiger, and C. Rother, "Augmented reality meets computer vision: Efficient data generation for urban driving scenes," *International Journal of Computer Vision*, vol. 126, no. 9, pp. 961–972, 2018.

[26] H. A. Alhaija, S. K. Mustikovela, A. Geiger, and C. Rother, "Geometric image synthesis," *arXiv preprint arXiv:1809.04696*, 2018.

[27] R. Khirodkar, D. Yoo, and K. M. Kitani, "VADRA: Visual adversarial domain randomization and augmentation," *arXiv preprint arXiv:1812.00491*, 2018.

[28] J. Fang, F. Yan, T. Zhao, F. Zhang, D. Zhou, R. Yang, Y. Ma, and L. Wang, "Simulating LiDAR point cloud for autonomous driving using real-world scenes and traffic flows," *arXiv preprint arXiv:1811.07112*, 2018.

[29] W. Li, C. Pan, R. Zhang, J. Ren, Y. Ma, J. Fang, F. Yan, Q. Geng, X. Huang, H. Gong *et al.*, "AADS: Augmented autonomous driving simulation using data-driven algorithms," *arXiv preprint arXiv:1901.07849*, 2019.

[30] D. J. Fremont, X. Yue, T. Dreossi, S. Ghosh, A. L. Sangiovanni-Vincentelli, and S. A. Seshia, "Scenic: Language-based scene generation," *CoRR*, vol. abs/1809.09310, 2018.

[31] Z. Liu, M. Shen, J. Zhang, S. Liu, H. Blasinski, T. Lian, and B. Wand ell, "A system for generating complex physically accurate sensor images for automotive applications," *arXiv e-prints*, p. arXiv:1902.04258, Feb 2019.

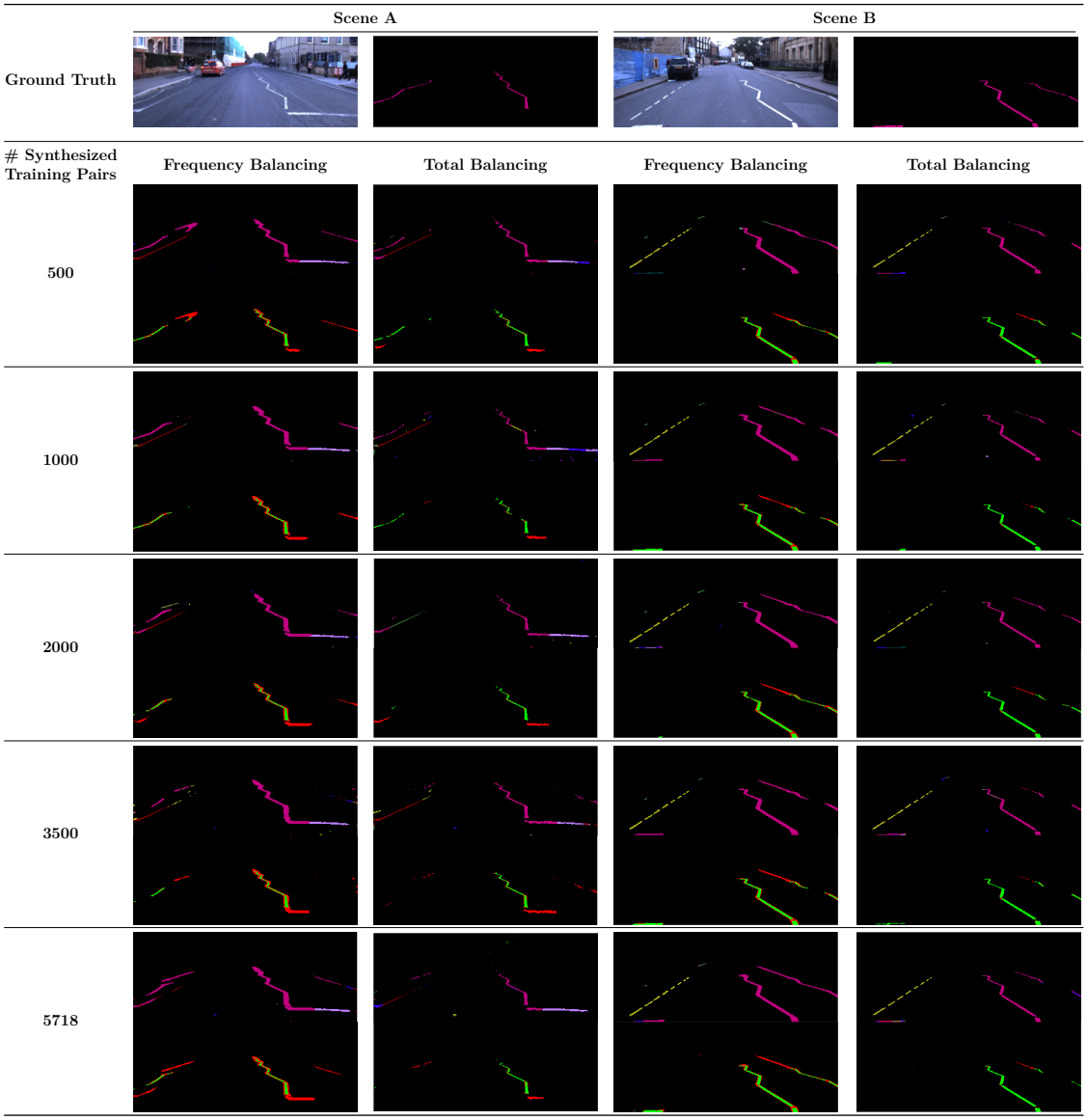
[32] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000 km: The Oxford Robotcar dataset," *The International Journal of Robotics Research*, vol. 36, no. 1, pp. 3–15, 2017.

[33] H. Porav, W. Maddern, and P. Newman, "Adversarial training for adverse conditions: Robust metric localisation using appearance transfer," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 1011–1018.

[34] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2015, pp. 234–241.

[35] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec 2015, pp. 2650–2658.

Table 6.1: A qualitative comparison of frequency and total balancing for the zig-zag road markings. The *top* row of every testing point displays the output prediction, the *bottom* row shows the false positive (*red*) and true positive (*green*) pixels. The number of false positive pixels increases with the number of synthetic training pairs for frequency balancing; total balancing alleviates this problem.



6.2.2 Further Details

The reproduced publication shows quantitatively that the newly-introduced cost function, which we referred to as *total balancing*, alleviates the precision drop that results from training with the frequency-balancing cost function. This problem arises when a large number of synthetic training pairs of a particular class are added

Table 6.2: Additional qualitative results for the bus stop road marking class, following the reproduced publication.

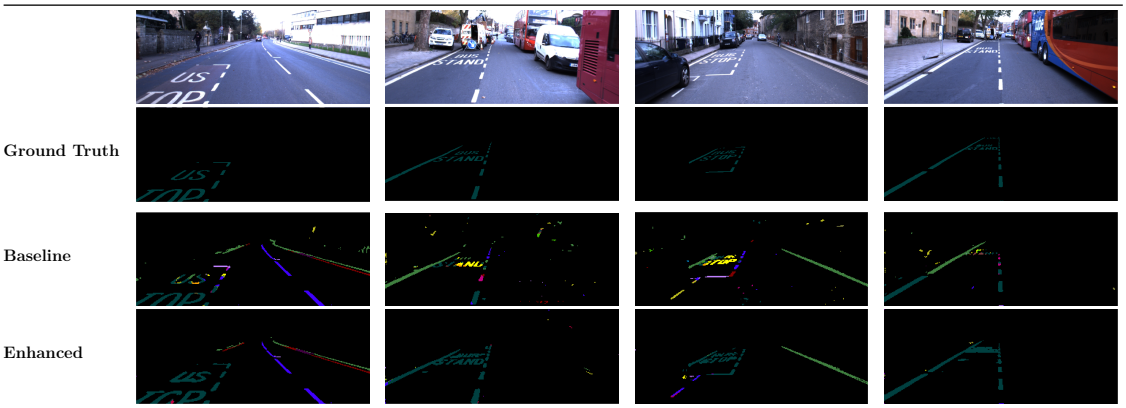
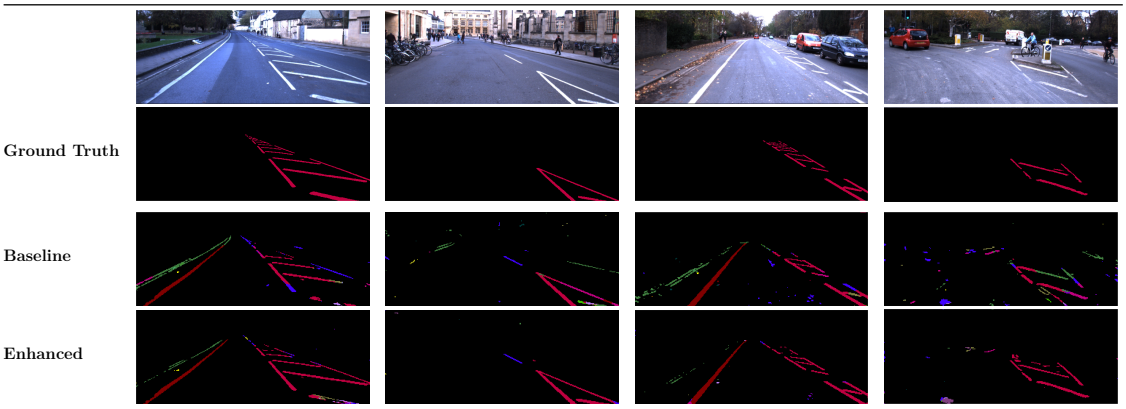
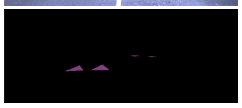
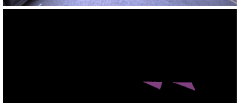
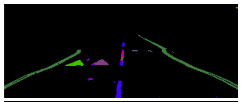
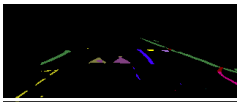
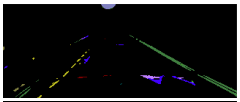
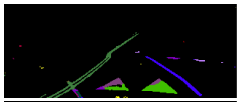
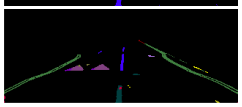
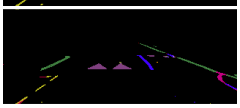
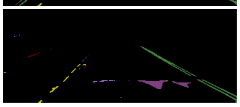
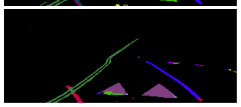


Table 6.3: Additional qualitative results for the diagonal road marking class, following the reproduced publication.



to a smaller (hand-labelled), more heterogeneous dataset as frequency balancing does not take this occurrence imbalance into account. Consequently, other road marking classes may be falsely classified as the synthesized class as they are now a minority class during training. We visualize this effect for the zig-zag road markings in Table 6.1. It is clear that frequency balancing leads to more false-positive pixels as the number of synthetic training pairs increases, thereby decreasing the precision. Total balancing appears to be more robust to this occurrence imbalance, showing similar performance in terms of the precision irrespective of the number of synthetic training pairs.

Table 6.4: Additional qualitative results for the warning triangle road marking class, following the reproduced publication.

				
Ground Truth				
Baseline				
Enhanced				

6.2.3 Further Results

We provide additional qualitative results for the rare classes studied in the publication in Table 6.2 - 6.5. It is clear that the enhanced model provides more consistent and correct segmentation of the rare classes.

Moreover, these scenes illustrate the difficulty of the task compared to the road scene segmentation investigated in Chapter 3. Road markings are much smaller than most of the scene classes, and there exists a higher degree of visual and geometrical similarity among the various road marking classes. Currently, state-of-the-art solutions for road marking classification achieve 40-45% mIoU, while the best methods for scene segmentation generally surpass 80% mIoU. In order to reach similar performance levels for the former task, alternative improvements are likely necessary besides better and more data. Possibilities include embedding priors regarding road marking construction directly into the network or changing the network architecture to handle smaller classes better. Furthermore, partial occlusions can easily lead to misclassifications, as discussed in the publication in Appendix C. These issues can be mitigated by tracking road markings over consecutive frames.

Table 6.5: Additional qualitative results for the zig-zag road marking class, following the reproduced publication.

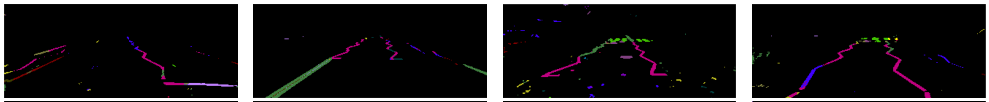
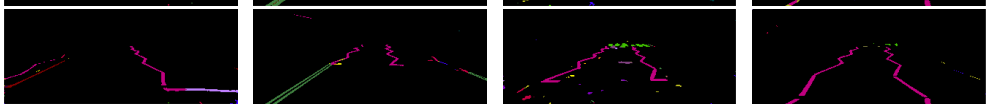
				
Ground Truth				
Baseline				
Enhanced				

Table 6.6: A comparison of different image-to-image translation frameworks for synthesizing bus stop road markings.

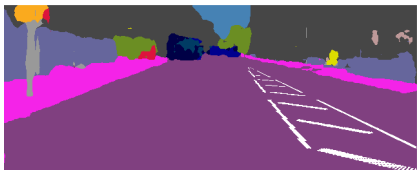
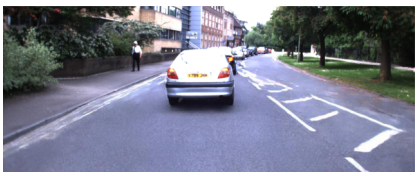
Semantic Label		
Pix2PixHD [68]		
SPADE [104]		

6.2.4 Further Discussion

As mentioned previously, synthesizing high-resolution images from semantic maps only became feasible recently with the introduction of the Pix2PixHD framework [68]. However, the normalization layers within its network attenuate semantic information [104]. It is, therefore, difficult for the network to synthesize road markings at a distance accurately. Instead, they become connected and blurred, losing their distinctive shape and size, which is crucial for correct classification.

The authors resolve this issue in the follow-up work [104] named SPADE. The semantic maps modulate the activations in the normalization layers through a

Table 6.7: A comparison of different image-to-image translation frameworks for synthesizing diagonal road markings.

Semantic Label		
Pix2PixHD [68]		
SPADE [104]		

spatially-adaptive learned transformation. In this way, small semantic details are consistently enforced throughout the layers of the generator resulting in more realistic road markings, especially at a farther distance. We demonstrate this improvement with bus stop and diagonal road markings in Table 6.6 and Table 6.7, respectively.

It is reasonable to expect that these results will improve further over time as computer vision researchers develop better frameworks for synthesizing images.

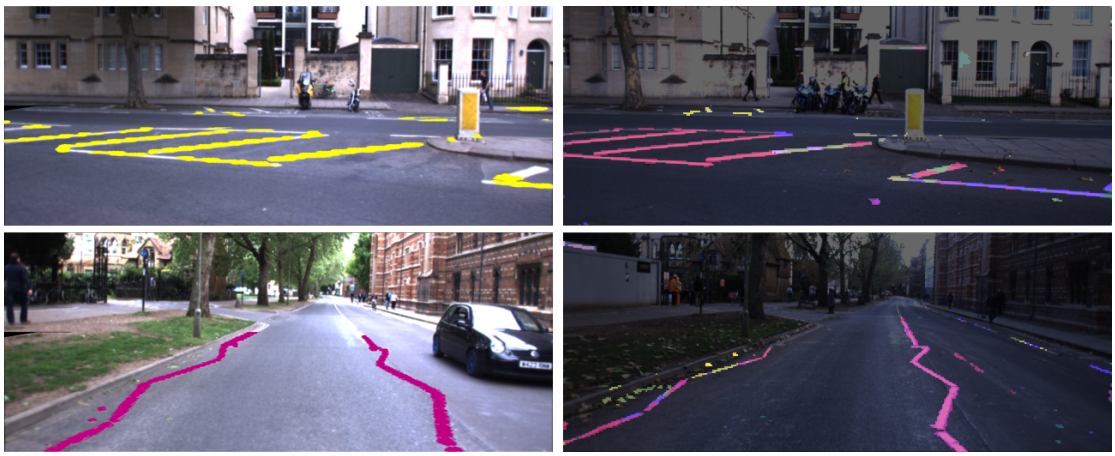
6.3 Qualitative Comparison

This section compares the model-driven and data-driven approach on several key aspects. We limit our comparison to qualitative examples as (1) the model-driven approach did not provide a quantitative evaluation and (2) the output representation of the two approaches differs substantially (i.e. a parametrised model versus a pixel-wise mask). We also provide some insights for combining the two approaches so that the acquired semantic road markings are accurate, complete, and can be integrated directly into the scene graph.

An advantage of both approaches is that they operate in a self-supervised way, requiring no hand-labelling (except for the initial semantic maps of the data-driven method, which were obtained from a pretrained Cityscapes model). Both approaches

(a) **(b)**

Figure 6.3: When the conditions are favourable (i.e. bright, newly-painted road markings on dark tarmac), the model-driven, **(a)**, and the data-driven approach, **(b)**, both accurately classify all the pixels belonging to the zig-zag road marking instance of interest.



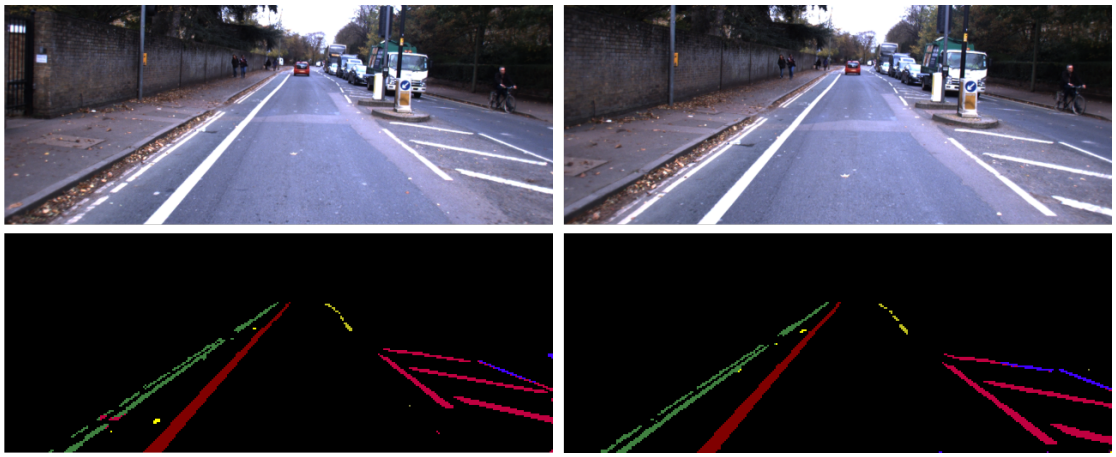
(a) **(b)**

Figure 6.4: Whereas the model-driven approach, **(a)**, outputs complete road marking instances (i.e. parametrised models), the data-driven approach, **(b)**, provides pixel-wise masks. For the latter, this might lead to spatial inconsistencies over a single road marking segment. The right segments of the diagonal markings (*top*, yellow in **(a)** and pink in **(b)**) are classified incorrectly, possibly because they are partially occluded by the traffic island and the image border. The pixels of the left zig-zag marking (*bottom*) are also classified inconsistently, possibly because it is slightly degraded and partially occluded by leaves.

achieve satisfactory results in favourable conditions (i.e. bright and non-degraded linear road markings on dark tarmac), as shown in Figure 6.3. However, the two approaches differ significantly in other aspects, as discussed below.

Output Representation

One crucial difference between the two approaches is the output representation. Whereas the model-driven approach outputs road marking instances (i.e. parametrised models) consisting of the classified pixels, the data-driven approach outputs a pixel-



(a) **(b)**

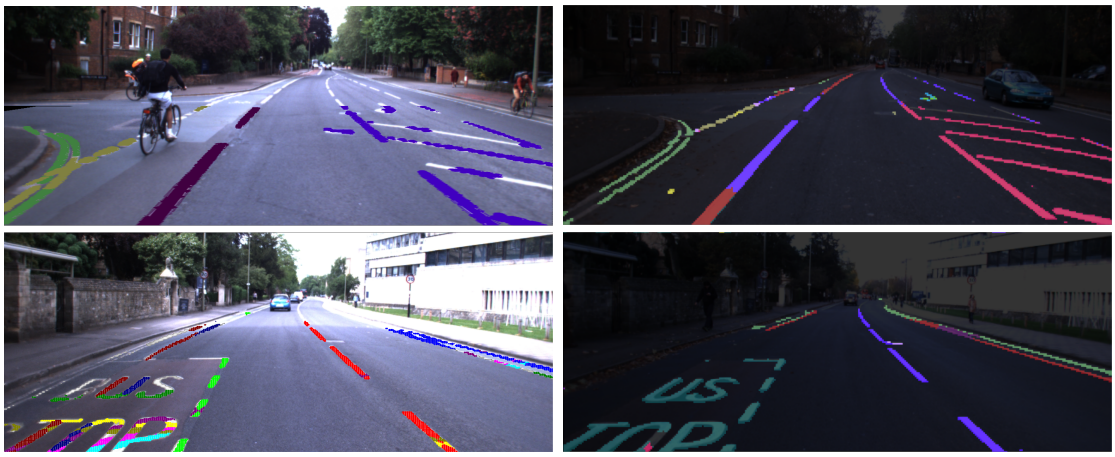
Figure 6.5: The data-driven approach does not track road markings over time, which decreases the performance. In the first frame, **(a)**, the diagonal road marking instance is classified almost entirely correctly. However, a few frames later, **(b)**, some segments have changed to an incorrect class. This issue can be resolved by changing the network architecture to include memory components.

wise mask. As the pixels are classified independently by the data-driven approach, this may lead to spatial inconsistencies (i.e. pixels of the same linear road marking segment classified differently). Two examples of this are given in Figure 6.4.

The ultimate purpose of a road marking classification system is to retrieve semantic instances distinguished by the road rules they convey. From a decision making perspective, there is no need to classify every road marking pixel correctly – as only the semantic meaning and location are relevant. With that in mind, the pixel-wise masks provided by the data-driven approach need to be post-processed [83] into semantic instances, which can be integrated efficiently and conveniently into the scene graph.

Tracking

As discussed in the publication in Appendix C, it is desirable to track road markings over consecutive frames. This improves temporal robustness and solves class ambiguity when road markings become partially occluded. The current data-driven approach is implemented on a per-frame basis (similar to the original U-Net) and thus does not exploit temporal cues. This decreases the performance, as exemplified



(a) **(b)**

Figure 6.6: The model-driven approach relies on the assumption that road markings are formed by collections of linear segments; therefore, it cannot classify arrows and letters, **(a)**. In contrast, the data-driven approach does not distinguish between those classes because it classifies pixels independently, **(b)**. The non-linear classes should ideally be classified by the network and combined with the linear road marking instances obtained by the model-driven approach.

in Figure 6.5. Fortunately, this issue can be resolved by changing the network architecture to incorporate memory components such as LSTMs.

Symbols & Letters

Certain road marking classes of interest are not formed by collections of linear segments (e.g. arrows, symbols, or letters). As this is the primary underlying assumption of the model-driven approach, it is not able to classify those classes correctly. In contrast, the data-driven approach works on a per-pixel basis; therefore, there is no distinction between letters and separators, for example. The difference between the two approaches is exemplified in Figure 6.6. The non-linear road marking pixels should ideally be classified by the network, clustered into instances, and combined with the semantic road marking instances retrieved by the model-driven approach.

6.4 Conclusion

This chapter compared a model-driven and a data-driven approach for road marking classification. Both approaches work in a self-supervised way by employing the binary road marking segmentation to reduce the required amount of manual labelling.

The model-driven approach retrieves linear road marking segments in the binary segmentation because most road marking classes are formed by collections of linear segments. A subsequent optimization step, constrained by road construction definitions, clusters the linear segments into their semantic road marking classes. It was demonstrated that this approach performs well under different conditions.

The data-driven approach synthesizes a training image for a predefined road marking label. More specifically, we alter the semantic map and use state-of-the-art image-to-image translation techniques to generate photo-realistic images. We have shown that this is an efficient method for generating large-scale datasets for different environments and that the quality of the synthesized images improves significantly as image-to-image translation frameworks progress. Furthermore, the synthesized datasets can be efficiently extended to multiple conditions with the techniques presented in Chapter 3. We have also introduced a new class-weighted loss function to boost the segmentation performance of rare road marking classes towards the state-of-the-art, although it converges after a certain number of synthetic training pairs. The most likely reason for this is that GANs suffer from mode collapse; hence, there is little additional information in the extra training pairs. Several works have studied generating informative images to improve beyond the initial convergence point. These future directions are discussed in Section 8.3.2.

We have compared the two approaches qualitatively to provide insights into some critical aspects for the real-world deployment of road marking classification systems. The ultimate goal is to retrieve the underlying meaning of the road markings in the scene, which serve as a basis for decision making. Whereas the data-driven approach outputs a pixel-wise mask that needs to be post-processed for this purpose, the model-driven approach outputs parametrized instances of semantic road marking classes, which can be integrated directly into the scene

graph. An advantage of the data-driven approach over the model-driven one is that it extends to symbols and letters because it does not rely on underlying linear models. Furthermore, we have shown the importance of tracking road markings over subsequent frames to improve temporal robustness and resolve ambiguity when road marking instances become partially occluded.

In conclusion, a complete road marking classification system combines aspects of the two presented approaches. Integrating the retrieved semantic road marking instances into the scene graph is left for further research.

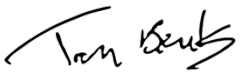
Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor **(only required where there isn't already a statement of contribution within the paper itself).**

Title of Paper	Generating All the Roads to Rome: Road Layout Randomization for Improved Road Marking Segmentation
Publication Status	Published
Publication Details	**T. Bruls**, H. Porav, L. Kunze, and P. Newman, "Generating all the roads to Rome: Road layout randomization for improved road marking segmentation", in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Oct. 2019, pp. 831-838.

Student Confirmation

Student Name:	Tom Adriaan Hubert Bruls		
Contribution to the Paper	All work except editorial changes and advice. Contributions included: - Generating the ideas. - Developing the software. - Preparing and processing the data. - Running the experiments. - Performing the analysis. - Writing the paper, creating the figures and tables.		
Signature		Date	10-05-2020

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Professor Paul Newman			
Supervisor comments			
Signature		Date	11-05-2020

This completed form should be included in the thesis, at the end of the relevant chapter.

7

Representations for Overview: Boosted Inverse Perspective Mapping

Contents

7.1 Publication 108
7.2 Further Details 117
7.3 Further Results 117
7.3.1 Boosted IPM 118
7.3.2 Road Marking Segmentation in Bird's-Eye View 124
7.4 Further Discussion 129
7.5 Conclusion 130

This chapter focusses on improving Inverse Perspective Mapping (IPM), another prerequisite for generating accurate scene graphs. As demonstrated in previous chapters, scene graphs are constructed from pixel-wise segmented road markings in the front-facing image to describe the road layout. These segmentations, together with the images, are generally transformed into a more convenient coordinate system (i.e. a view) in order to be utilized effectively [30], [80]. This transformation is commonly referred to as IPM. IPM takes the frontal view as an input and applies a homography transformation, which maps the pixels to a different 2D coordinate frame, to produce a top-down view of the scene known as a *bird's-eye view*. Many tasks in the autonomous driving pipeline benefit from this transformation because it

changes the image perspective so that an object should appear the same irrespective of its position in the IPM image, making registration and reasoning easier.

Traditional homography-based IPM relies on three assumptions: (1) the camera is in a fixed position with respect to the road, (2) the road surface is planar, and (3) the road surface is free of obstacles. A violation of any one of these assumptions will degrade the quality of the IPM image [89], [107], [108]. Another significant drawback arises from the fact that distant objects occupy smaller pixel regions in the front-facing image. These pixels are interpolated in order to create a dense bird's-eye-view image, leading to an unnatural blurring and stretching of farther-away objects such as road markings [91]. Hence, these objects are more difficult to segment and no longer adhere to the spatial and relational properties defined by road construction definitions. We have shown in Section 4.2.2 that such (minor) inaccuracies can lead to significant qualitative differences in the semantic interpretation of a scene.

We have published an alternative adversarial learning framework, which produces a significantly improved IPM, called *boosted IPM*, online from a single front-facing camera image to overcome these limitations. The case for a learning framework is supported by the fact that although the IPM transformation may seem complex, it is not random. It is, in fact, structured but non-linear, and therefore this mapping can be learned with a DNN.

The authors of [109] proposed the first generative learning approach for IPM simultaneously with the publication of our work. In contrast to our work, which demonstrates benefits for real-world data, they use synthetic data and learn to generate a bird's-eye view only for a small region surrounding the ego vehicle, which is not sufficient for the type of scene understanding we aim to achieve in this thesis. The authors of [91] later extended the ideas presented in our publication to directly improve the output task (in their case lane detection) instead of optimising for the most realistic bird's-eye view.

Our publication is reproduced in Section 7.1. It demonstrates the use of a new type of network architecture, the Incremental Spatial Transformer GAN, which learns the extensive appearance transformation between the front-facing

and bird's-eye view incrementally. The network is trained in a self-supervised way as the training data is generated automatically from the sensor calibrations and VO. Boosted IPM leads to sharper road elements and a more homogeneous illumination during inference while it allows for reasoning about the road layout of occluded areas. Therefore, it is beneficial for scene understanding tasks such as scene graph generation, and we demonstrate in the publication that it resolves some of the aforementioned limitations. We expand upon the publication in the remaining sections of this chapter.

In summary, this chapter makes the following principal contributions:

- A learned, improved alternative to the commonly-used homography-based IPM for autonomous vehicles, *boosted IPM*, and its derivative, *boosted RL* (T-1c).

- A method for generating data for training boosted IPM in a self-supervised way from sensor calibrations and VO (T-2a).

- An evaluation of various extensions of boosted IPM and road marking segmentation therein by using the CARLA driving simulator (T-1b and T-1c).

Additionally, this chapter makes the following supporting contributions in collaboration:

- A new network architecture, the Incremental Spatial Transformer GAN, for learning image-to-image transformations with large appearance changes. The network generates boosted IPM online from a single front-facing image under different conditions (T-1c and T-3b).

- A demonstration of the advantages of boosted IPM for road marking segmentation and the semantic interpretation of real-world road scenes, possibly in the presence of occlusions and extreme illumination (T-1a, T-1b, and T-1c).

7.1 Publication

This section contains a reproduction of the following publication:

> [21] **T. Bruls***, H. Porav*, L. Kunze, and P. Newman, "The right (angled) perspective: Improving the understanding of road scenes using boosted inverse perspective mapping", in *Proceedings of the Intelligent Vehicles Symposium (IV)*, Jun. 2019, pp. 302-309[1].

[1] video accompanying the publication with extensive explanations and results: `https://www.youtube.com/watch?v=JLOAayZe1Do`

The Right (Angled) Perspective: Improving the Understanding of Road Scenes Using Boosted Inverse Perspective Mapping

Tom Bruls*, Horia Porav*, Lars Kunze, and Paul Newman

***Abstract*— Many tasks performed by autonomous vehicles such as road marking detection, object tracking, and path planning are simpler in bird's-eye view. Hence, Inverse Perspective Mapping (IPM) is often applied to remove the perspective effect from a vehicle's front-facing camera and to remap its images into a 2D domain, resulting in a top-down view. Unfortunately, however, this leads to unnatural blurring and stretching of objects at further distance, due to the resolution of the camera, limiting applicability. In this paper, we present an adversarial learning approach for generating a significantly improved IPM from a single camera image in real time. The generated bird's-eye-view images contain sharper features (e.g. road markings) and a more homogeneous illumination, while (dynamic) objects are automatically removed from the scene, thus revealing the underlying road layout in an improved fashion. We demonstrate our framework using real-world data from the Oxford RobotCar Dataset and show that scene understanding tasks directly benefit from our boosted IPM approach.**

I. Introduction

Autonomous vehicles need to perceive and fully understand their environment to accomplish their navigation tasks. Hence, scene understanding is a critical component within their perception pipeline, not only for navigation and planning, but also for safety purposes. While vehicles use different types of sensors to interpret scenes, cameras are one of the most popular sensing modalities in the field, due to their low cost as well as the availability of well-established image processing techniques.

In recent years, deep learning approaches based on images have been very successful and significantly improved the performance of autonomous vehicles in the context of semantic scene understanding [1], [2]. Many of these approaches take images from a front-facing camera as their input. However, images as well as their interpretations (i.e. segmented pixels) in this perspective are often transformed into a local and/or global coordinate system (or view) to be utilized effectively within tasks such as lane detection [3], [4], road marking detection [5], road topology detection [6], [7], object detection/tracking [8]–[10], as well as path planning and intersection prediction [11], [12]. This transformation is commonly referred to as Inverse Perspective Mapping (IPM) [13]. IPM takes the frontal view as input, applies a homography, and produces a top-down view of the scene by mapping the pixels to a different 2D-coordinate frame, which is also known as *bird's-eye view*.

∗ equal contribution

Authors are from the Oxford Robotics Institute, Dept. Engineering Science, University of Oxford, UK. {tombruls, horia, lars, pnewman}@robots.ox.ac.uk

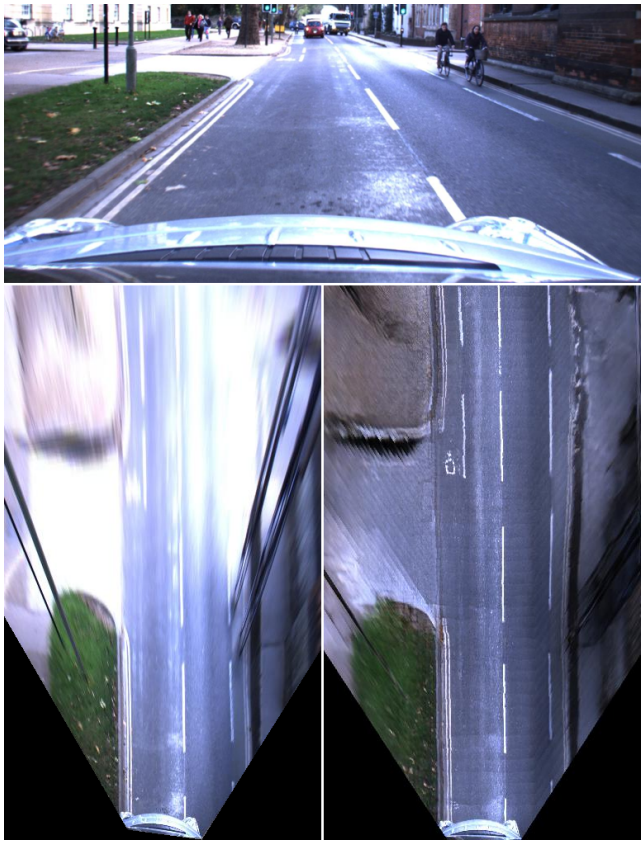


Fig. 1. Boosted Inverse Perspective Mapping (IPM) to improve the understanding of road scenes. *Left:* Top-down view created by applying a homography-based IPM to the front-facing image (*top*), leading to unnatural blurring and stretching of objects at further distance. *Right:* Improved top-down view generated by our Incremental Spatial Transformer GAN, containing sharper features and a homogeneous illumination, while dynamic objects (i.e. the two cyclists) are automatically removed from the scene.

In practice, IPM works well in the immediate proximity of the vehicle (assuming the road surface is planar). However, the geometric properties of objects in the distance are affected unnaturally by this non-homogeneous mapping, as shown in Fig. 1. This limits the performance of applications in terms of their accuracy and the distance at which they can be applied reliably. More crucial, however, is the effect of inaccurate mappings on the semantic interpretation of scenes, where small inaccuracies can lead to significant qualitative differences. As we demonstrate in Section V-B (Table I), these qualitative differences can manifest themselves in many ways, including missing lanes and/or late detection of stop lines (or other critical road markings).

To overcome these challenges, we present an adversarial learning approach which produces a significantly improved IPM in real time from a single front-facing camera image. This is a difficult problem which is not solved by existing methods, due to the large difference in appearance between the frontal view and IPM. State-of-the-art approaches for cross-domain image translation tasks train (conditional) Generative Adversarial Networks (GANs) to transform images to a new domain [14], [15]. However, these methods are designed to perform aligned appearance transformations and struggle when views change drastically [16]. The latter work, in which a synthetic dataset with *perfect* ground-truth labels is used to learn IPM, is closest to ours.

We demonstrate in this paper that we are able to generate reliable, improved IPM for larger scenes than in [16], which are therefore able to directly aid scene understanding tasks. We achieve this in real time using real-world data collected under different conditions with a single front-facing camera. Consequently, we must deal with *imperfect* training labels (see Section IV) created from a sequence of images and ego-motion. An Incremental Spatial Transformer GAN is introduced to address the significant appearance change between the frontal view and IPM. Compared to analytic IPM approaches our learned model is (1) more realistic with sharper contours at long distance, (2) invariant to extreme illumination under different conditions, and (3) removes dynamic objects from the scene to recover the underlying road layout. We make the following contributions in this paper:

- we introduce an Incremental Spatial Transformer GAN for generating boosted IPM in real time;
- we explain how to create a dataset for training IPM methods on real-world images under different conditions; and
- we demonstrate that our boosted IPM approach improves the detection of road markings as well as the semantic interpretation of road scenes in the presence of occlusions and/or extreme illumination.

II. Related Work

Improved IPM As indicated in Section I, many applications can be found in the literature that apply IPM. They rely on three assumptions: (1) the camera is in a fixed position with respect to the road, (2) the road surface is planar, and (3) the road surface is free of obstacles. Remarkably, relatively few approaches exist that aim to improve inaccurate IPM, in case one or more of these assumptions are not satisfied.

Several works have tried to adjust for inaccuracies caused by invalidity of the first two assumptions. The authors of [17], [18] used vanishing point detection, [19] estimated the slope of the road according to the lane markings, and [20] employed motion estimation obtained from SLAM. Invalidity of the third assumption is tackled in [21] by using a laser scanner to exclude obstacles from being transformed to IPM. Another approach [22]–[24] creates a look up table for all pixels, by taking into account the distance of objects on the road surface, in order to reduce artefacts at further distance. However, these methods generally assume simple environments (i.e. highway). Contrarily, we learn a non-linear mapping more suited for urban scenes.

Very recently, [16] proposed the first learning approach for IPM using a synthetic dataset. The authors introduced BridgeGAN which employs the homography IPM to bridge the significant appearance gap between the frontal view and bird's-eye view. In contrast, we use real-world data and consequently *imperfect* labels to generate boosted IPM for larger scenes. Therefore, our learned mapping is directly beneficial for scene understanding tasks (see Section V-B).

Semantic IPM Several methods use the semantic relations between the two views for different tasks. In [25], [26] conditional random fields in the frontal view and IPM are optimized to retrieve a coarse semantic bird's-eye-view map from a sequence of camera images. A joint optimization net is trained in [27], [28] to align the semantic cues of the two views. The authors then train a GAN to synthesize a ground-level panorama from the coarse semantic segmentation. However, because aerial images differ significantly in appearance from the ground view, there is a lack of texture and detail in the synthesized images. We generate a more detailed IPM by learning a direct mapping of the pixels from the frontal view which is more useful for autonomous driving applications.

GANs for Novel View Synthesis The rise of GANs has made it possible to generate new, realistic images from a learned distribution. In order to guide the generation process towards a desired output, GANs can be conditioned on an input image [14], [29]. Until now, these methods were restricted to perform aligned appearance transformations.

In [30], the spatial transformer module was introduced to learn transformations of the input to improve classification tasks. The authors of [31], [32] used similar ideas to synthesize new views of 3D objects or scenes. More recently, these two fields were combined in [33], [34]. In the latter work, realistic compositions of objects are generated for a new viewpoint. However, these techniques are limited to toy datasets or distort real-world scenes with dynamic objects.

III. Boosted IPM using an Incremental Spatial Transformer GAN

A. Network Overview

As a starting point, we use a state-of-the-art architecture similar to the global enhancer of [29], without employing boundary or instance maps. Additionally, as we expect a slight change in scale from the homography-based IPM image to the stitched training labels (see Section IV), we refrain from using any pixel-wise losses and instead use multi-scale discriminator losses [29] combined with a perceptual loss [35], [36] based on VGG16 [37]. While VGG16 is trained on the ImageNet [38] dataset, thus being more suitable for frontal rather than bird's-eye-view images of road scenes, we still leverage the stability of its encoded features in this study. Retraining VGG16 on bird's-eye-view images of road scenes or swapping it out for a more suitable model, may improve the quality of the generated images, but this is beyond the scope of this study.

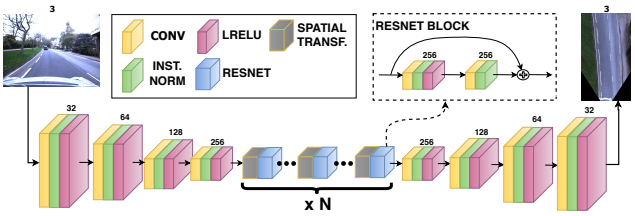


Fig. 2. The architecture of the generator of the network. The bottleneck of the model contains a series of N sequential blocks. Each block performs an incremental perspective transformation of n degrees, so that the bottleneck as a whole transforms the features from frontal to bird's-eye view. After every transformation, the features are sharpened by a ResNet block before the next transformation is applied. This process is depicted in more detail in Fig. 3.

Our model follows a largely traditional downsample-bottleneck-upsample architecture, where we reformulate the bottleneck portion of the model as a series of N_{STRes} blocks that perform incremental perspective transformations followed by feature enhancement. Each block contains a Spatial Transformer (ST) [30] followed by a ResNet layer [39]. The structure of the generator is presented in Fig. 2. For an in-depth description of the remaining architecture, the reader is directed towards the paper and supplemental material of [29].

B. Spatial ResNet Transformer

Since far-away real-world features are represented by a smaller pixel area as compared to identical close-by features, a direct consequence of applying a full perspective transformation to the input is increased unnatural blurring and stretching of the features at further distance. To counteract this effect, our model divides the full perspective transformation into a series of N_{STRes} smaller incremental perspective transformations, each followed by a refinement of the transformed feature space using a ResNet block [39]. The intuition behind this is that the slight blurring that occurs as a result of each perspective transformation is restored by the ResNet block that follows it, as conceptually visualized in Fig. 3. To maintain the ability to train our model end-to-end, we apply these incremental transforms using Spatial Transformers [30].

Intuitively, a Spatial Transformer is a mechanism, which can be integrated in a deep-learning pipeline, that warps an image using a parametrization (e.g. an affine or homography transformation matrix) conditioned on a specific input signal. Formally, each incremental spatial transformer is an end-to-end differentiable sampler, represented in our case by two major components:

- a convolutional network which receives an input I of size $H_I * W_I * C$, where H_I, W_I and C represent the height, width, and number of channels of the input respectively, and outputs a parametrization M_{loc} of a perspective transformation of size $3 * 3$, and;
- a Grid Sampler which takes I and M_{loc} as inputs, creates a mapping matrix M_{map} of size $H_O * W_O * 2$, where H_O and W_O represent the height and width of the output O. M_{map} maps homogeneous coordinates $[x, y, 1]^T$ to their new warped position given by $M_{\text{loc}} * [x, y, 1]^T$. Finally, M_{map} is used to construct O in the following way: $O(x, y) = I(M_{\text{map}}(x, y, 1), M_{\text{map}}(x, y, 2))$.

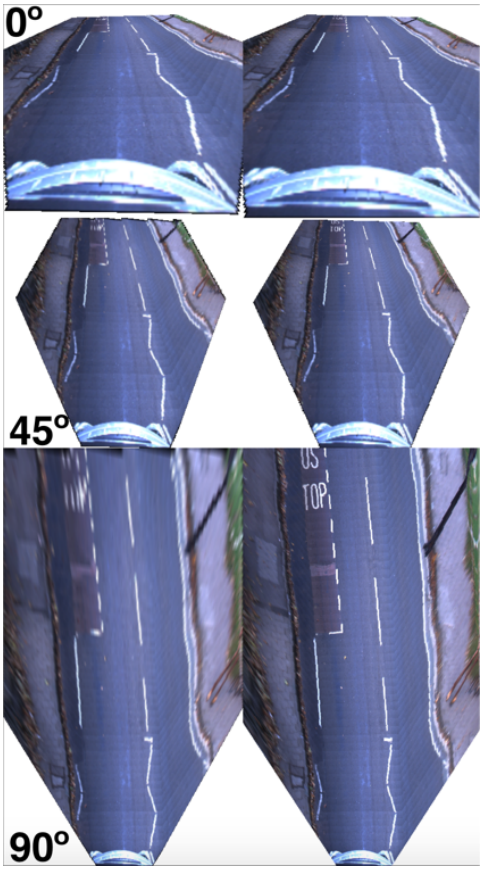


Fig. 3. Conceptual visualization of the sequential incremental transformations (i.e. $N = 3$, from $0°$ to $90°$ degrees down the rows) occurring in the bottleneck of the generator. The left column shows the features immediately after the transformation is applied, consequently they are stretched and blurred (e.g. BUS STOP letters). The right column shows how the ResNet blocks learns to sharpen these features to create the improved IPM before the next transformation is applied. Note that in reality the bottleneck has 512 feature maps instead of the 3 RGB channels depicted here for demonstration purposes.

In practice, it is non-trivial to train a spatial transformer (and even less trivial; a sequence of spatial transformers) on inputs with a large degree of self-similarity, such as road scenes. To stabilize the training procedure, for each incremental spatial transformer, we decompose $M_{\text{loc}} = M_{\text{locref}} * M_{\text{locpert}}$, where M_{locref} is initialized with an approximate parametrization of the desired incremental homography, and M_{locpert} is the actual output of the convolutional network and represents a learned perturbation or refinement of M_{locref}.

C. Losses

Our architecture stems from [29], but does not make use of any instance maps. Due to the potential misalignment between the output of the network and the labels (see Section IV), we rely on a multi-scale discriminator loss and a perceptual loss based on VGG16. With a generator G,

k^{th} scale discriminator D_k, and $\mathcal{L}_{\text{GAN}}(G, D_k)$ being the traditional GAN loss defined over $k = 3$ scales as in [29], the final objective thus becomes:

$$\mathcal{L}_{\text{tot}} = \min_G((\max_{D_1,D_2,D_3} \sum_{k=1,2,3} \mathcal{L}_{\text{GAN}}(G, D_k)) + \lambda_{\text{FM}} \sum_{k=1,2,3} \mathcal{L}_{\text{FM}}(G, D_k) + \lambda_{\text{VGG}} \mathcal{L}_{\text{VGG}}(G)), \tag{1}$$

where $\mathcal{L}_{\text{FM}}(G, D_k)$ is the multi-scale discriminator loss:

$$\mathcal{L}_{\text{FM}(G,D_k)} = \sum_{i=1}^{l_D} \frac{1}{w_i} \| D_k(I_{\text{label}})_i - D_k(G(I_{\text{input}}))_i \|_1, \tag{2}$$

and $\mathcal{L}_{\text{VGG}}(G)$ is the perceptual loss:

$$\mathcal{L}_{\text{VGG}(G)} = \sum_{i=1}^{l_P} \frac{1}{w_i} \| \text{VGG}(I_{\text{label}})_i - \text{VGG}(G(I_{\text{input}}))_i \|_1, \tag{3}$$

with l_D denoting the number of discriminator layers used in the discriminator loss, l_P denoting the number of layers from VGG16 that are utilized in the perceptual loss, and I_{input} and I_{label} being the input and label images, respectively. The weights $w_i = 2^{l-i}$ are used to scale the importance of each layer used in the loss.

D. Implementation details

We choose $N_{\text{STRes}} = 6$, $N_{\text{downsample}} = 4$, $N_{\text{upsample}} = 4$ and $l_D = l_P = 4$. Furthermore, for training, we employ the Adam solver using a base learning rate set at 0.0002, and a batch size of 1, training for 200 epochs. For the loss trade-off, we empirically set $\lambda_{\text{FM}} = 5$ and $\lambda_{\text{VGG}} = 2$. We train our network using 8416 overcast and 4894 nighttime labels. At run time, the network performs inference in real time (≈ 20 Hz) using an NVIDIA TITAN X.

IV. Creating Training Data for Boosted IPM

To evaluate our approach, we use the Oxford RobotCar Dataset [40], which features a 10-km route through urban environments under different weather and lighting conditions.

In order to create training labels which are a better representation of the real world than the standard, homography-based IPM, we use a sequence of images from the front-facing camera and corresponding visual odometry [41], and merge them into a single bird's-eye-view image.

From the sensor calibrations and the camera's intrinsic parameters, we compute the transformation which defines the one-to-one mapping between the pixels of the front-facing camera and the bird's-eye view. Then, using the relative transform obtained by visual odometry between the current image frame of the sequence and the initial frame, we stitch the respective pixels of the current frame into the IPM image at the correct pixel positions. This operation is performed iteratively, overwriting previous IPM pixels with more accurate pixels of subsequent frames, until the vehicle has reached the end of its field of view of the initial image.

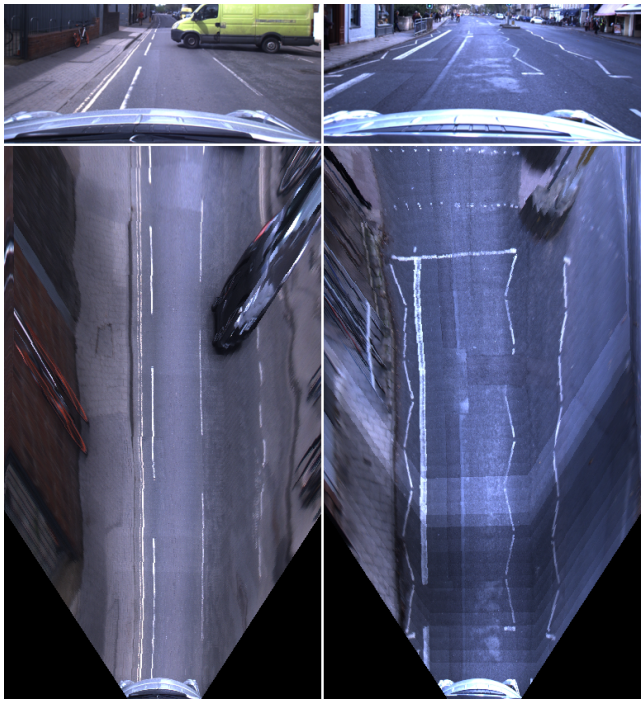


Fig. 4. Examples of created training pairs (which show the difficulties of using real-world data) by stitching IPM images generated from future front-facing camera images using the ego-motion obtained from visual odometry. The *left* example illustrates (1) movement of dynamic objects by the time the images are stitched and (2) stretching of objects because they are assumed to be on the road surface. The *right* example shows a significant change of illumination conditions. *Both* show inaccuracies at further lateral distance (e.g. wavy curb) because of sloping road surface and possibly imprecise motion estimation.

As the training labels are created from real-world data (in contrast to the synthetic data of [16]), their quality is limited by several aspects (see examples in Fig. 4):

- Minor inaccuracies in the estimation of the rotation of the vehicle and sloping road surface can lead to imprecise stitching at further lateral distance.
- Consecutive image frames may vary significantly in terms of lighting (e.g. due to overexposure), leading to illumination differences in the label which do not naturally occur in the real-world.
- Dynamic objects in the front-facing view will appear in a different position in future frames. Consequently, they will appear in unexpected places in the label.
- Objects above the road plane (e.g. vehicles, bicyclists, intersection islands, etc.) undergo a large deformation due to the view transformation. We cannot obtain accurate labels for these in real-world scenarios.

Due to the aforementioned drawbacks, no direct relation exists between the output (boosted IPM) of our network and the stitched labels. Therefore, it is impossible to incorporate a direct pixel-wise loss function, or employ super-resolution generating networks such as [42]. On the other hand, since we use a sequence of future images, regions that were previously occluded by (dynamic) objects in the initial view are potentially revealed later. This gives the network the

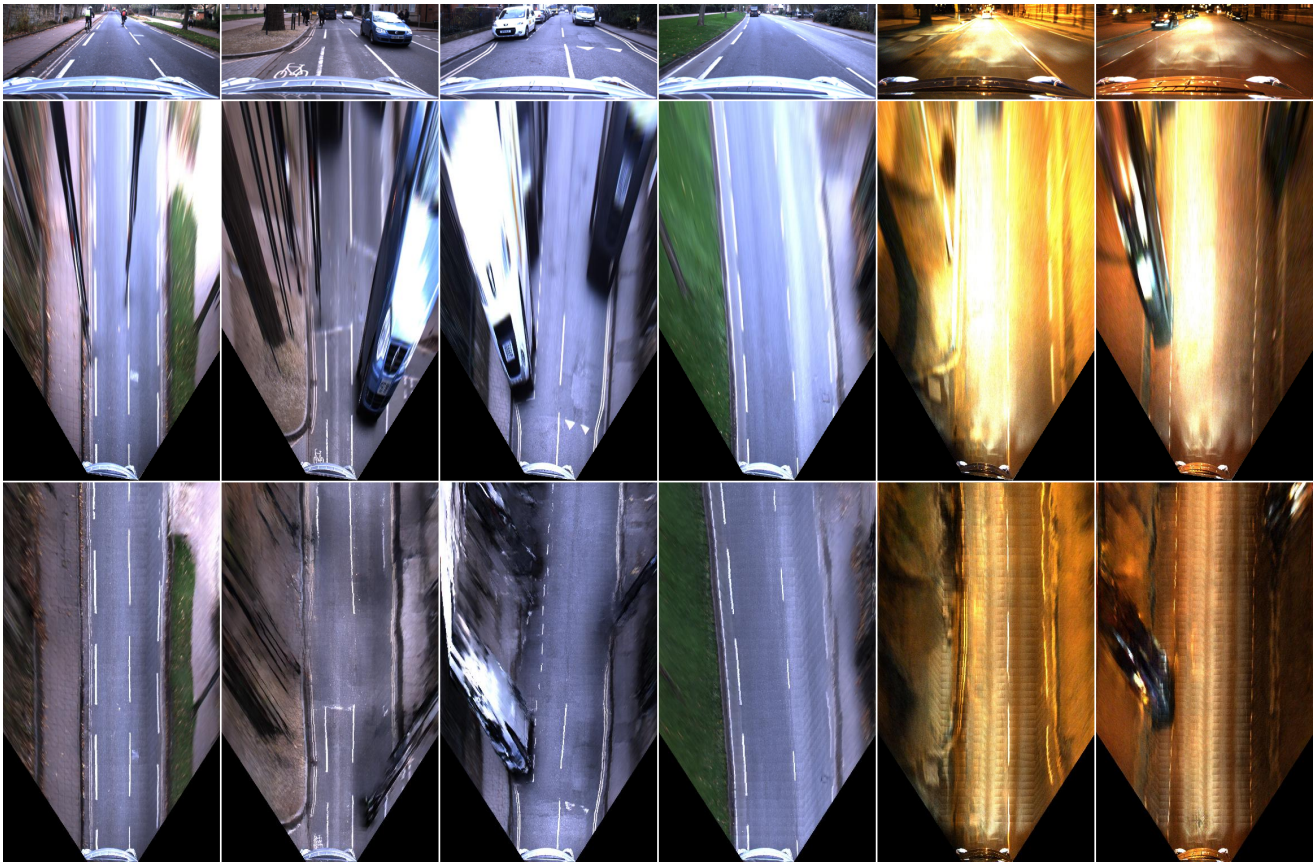


Fig. 5. Boosted IPM generated by the network (*bottom*) under different conditions compared to traditional IPM generated by applying a homography (*middle*) to the front-facing camera image (*top*). The boosted birds-eye-view images contain sharper features (e.g. road markings), more homogeneous illumination, and automatically remove (dynamic) objects from the scene. Consequently, we infer the underlying road layout, which is directly beneficial for various tasks performed by autonomous vehicles.

ability to learn the underlying road layout irrespective of occlusions or extreme illumination.

V. Experimental Results

In this section we present qualitative results generated under different conditions. Due to the nature of the problem, it is extremely hard to capture ground-truth labels in the real world (see Section IV), and thus to present quantitative results for our approach. Furthermore, the synthetic dataset used in [16] is not publicly available. However, we demonstrate that our boosted IPM has a significant qualitative effect on the semantic interpretation of real-world scenes. Lastly, we show some limitations of the presented framework.

A. Qualitative Evaluation

Fig. 5 shows qualitative results on a RobotCar test dataset. The results demonstrate that the network has learned the underlying road layout of various urban traffic scenarios. Semantic road features such as parking boxes (i.e. small separators) and stop lines are inferred correctly. Furthermore, dynamic objects, which occlude parts of the scene, are removed and replaced by the correct road/lane boundaries, making the representation more suitable for scene understanding and planning. The boosted IPM contains sharper road markings, which improves the performance of tasks such as lane detection. Lastly, the new view offers a more homogeneous illumination of the road surface, which is beneficial for all tasks that require image processing.

Additionally, we show that our framework is not limited to datasets recorded under overcast conditions. Although artificial lighting during nighttime introduces artefacts in the output, we are still able to significantly improve the representation of the underlying layout of the scene.

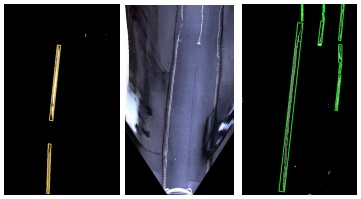
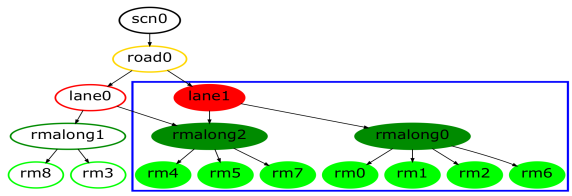
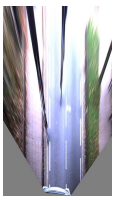
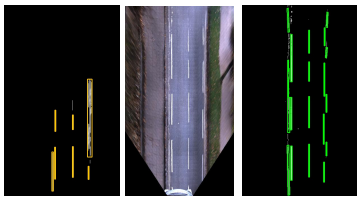
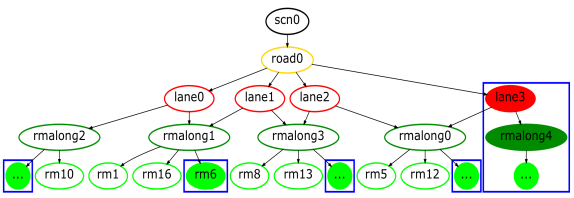
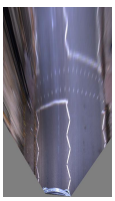
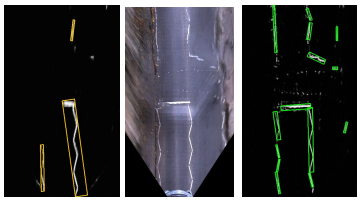
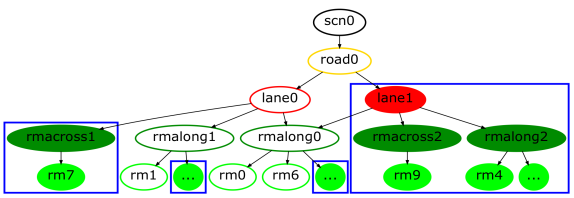
B. Employing Boosted IPM for Scene Interpretation

We demonstrate the effectiveness of our improved IPM approach for the application of road marking detection [43] and scene interpretation [44] (cf. Table I). Table I shows the original front-facing camera image, the bird's-eye views (homography-based as well as our boosted IPM) and their corresponding road marking detections, and the generated graph-based scene description.

The input to the scene interpretation process is the binary image mask of the detected road markings. Within these experiments this input is either provided by the homography-based IPM or by our boosted IPM. We then cluster the road marking pixels into groups and compute a set of spatial

TABLE I

QUALITATIVE EFFECTS OF IPM METHODS ON ROADMARKING DETECTION AND SCENE INTERPRETATION

Original	**Road marking Detection [43]** Homography	Boosted IPM	**Scene Interpretation [44]** (generated from detected road markings)
			
			
			

properties and relations. Based on the spatial information and a learned probabilistic grammar, which captures the road layout of scenes, a hierarchical, graph-based scene description is generated including information about roads, lanes and road markings (which are grounded in image space). The reader is directed towards [44] for more details.

As the overall scene interpretation is based on the segmentation of road markings, the quality of the road marking detection has a major impact on the generated scene graph, as demonstrated later. Experimentally, we have verified that boosted IPM allows us to more robustly detect road markings (1) at greater distance and (2) in more detail, and (3) infer road markings occluded by dynamic objects such as cars and cyclists. These improvements are possible because boosted IPM contains sharper features with more consistent geometric properties (at further distance) and learns the underlying road layout.

We have trained a road marking detection network for each view separately (because we expect a difference in learned features) with an equivalent setup according to [43]. Labels (in the front-facing image) were generated automatically by using the techniques of [43] and mapped down into IPM to match the input images. In addition, the boosted IPM road marking labels were stitched similarly to the camera images. Although the labels are not equivalent to the ground-truth, they have proven to be sufficient for training purposes if regularization techniques are applied. The increase in performance for road marking detection in the boosted IPM has immediate consequences for the interpretation of scenes. In general, all interpretations (scene graphs) benefit from more accurate road marking detection. Table I depicts qualitative differences in the scene graphs[1]. In the following we discuss the individual scenes.

Scene (A) The vehicle approaches a pedestrian crossing which is signaled by the upcoming zig-zag lines (visible at the top of the image). While these road markings are visible to the human eye in the homography-based IPM, the trained road marking detection network was not able to detect them because of the stretching and blurring at further distance. However, our boosted IPM produced a bird's-eye-view image with sharper contours for the zig-zag lines and correct reconstruction of the road markings occluded by the vehicle. This resulted in an improved scene graph which not only captured the right boundary of the ego lane, but also a previously undetected second lane on the right. Such qualitative differences have substantial impact on the planning and decision making of the vehicle.

Scene (B) The vehicle drives on a road with four lanes — two inner lanes for vehicles and two outer lanes for cyclists — and experiences a sudden change in illumination (from a darker foreground to a brighter background). This is clearly visible in the homography-based IPM and consequently leads to a poor detection of road markings. In contrast, our boosted approach produces a top-down view which inpaints learned semantic cues (i.e. road markings) directly over the overexposed area and also excludes the two cyclists. Hence, the resulting scene graph captures more

[1]In the scene graphs, the qualitative differences resulting from our boosted IPM method are indicated by filled nodes grouped in blue boxes.

detail as well as an extra lane which was missed in the segmentation resulting from the standard approach.

Scene (C) The vehicle approaches a pedestrian crossing which is indicated by both zig-zag and stop lines. Again, the distorted and blurry image resulting from the homography-based IPM leads to a poor detection of road markings. Our boosted approach has generated a more detailed view which led to better road marking detection including the successful identification of the stop lines. The resulting scene graph based on the homography-based IPM not only misses a lane, but crucially also both stop lines.

Such qualitative differences clearly demonstrate the advantage of our proposed method as they have a direct impact on planning and decision making of autonomous vehicles. While the detection and interpretation of road markings at a greater distance will enable an autonomous vehicle to adapt its behaviour earlier, the detection of road markings behind moving objects will lead to performance that is more robust and safer even when the scene is partly occluded.

C. Failure Cases

Under certain conditions, the boosted IPM does not accurately depict all details of the bird's-eye view of the scene.

As we cannot enforce a pixel-wise loss during training (Section IV), the shape of certain road markings is not accurately reflected (illustrated in Fig. 6). Improvement of the representation of these structural elements will be investigated in future work.

Furthermore, the spatial transformer blocks assume that the road surface is more or less planar (and perpendicular to the z-axis of the vehicle). When this assumption is not satisfied, the network is unable to accurately reflect the top-down scene at further distance. This might be solved by providing/learning the rotation of the road surface with respect to the vehicle.

VI. Conclusion

We have presented an adversarial learning approach for generating boosted IPM from a single front-facing camera image in real time. The generated results show sharper features and a more homogeneous illumination, while (dynamic) objects are automatically removed from the scene. Overall, we infer the underlying road layout, which is directly beneficial for tasks performed by autonomous vehicles such as road marking detection, object tracking, and path planning.

In contrast to existing approaches, we used real-world data collected under different conditions, which introduced additional issues due to varying illumination and (dynamic) objects, making it impossible to employ a pixel-wise loss during training. We have addressed the significant appearance change between the views by introducing an Incremental Spatial Transformer GAN.

We have demonstrated reliable, qualitative results in different environments and under varying lighting conditions. Furthermore, we have shown that the boosted IPM view allows for improved hierarchical scene understanding.

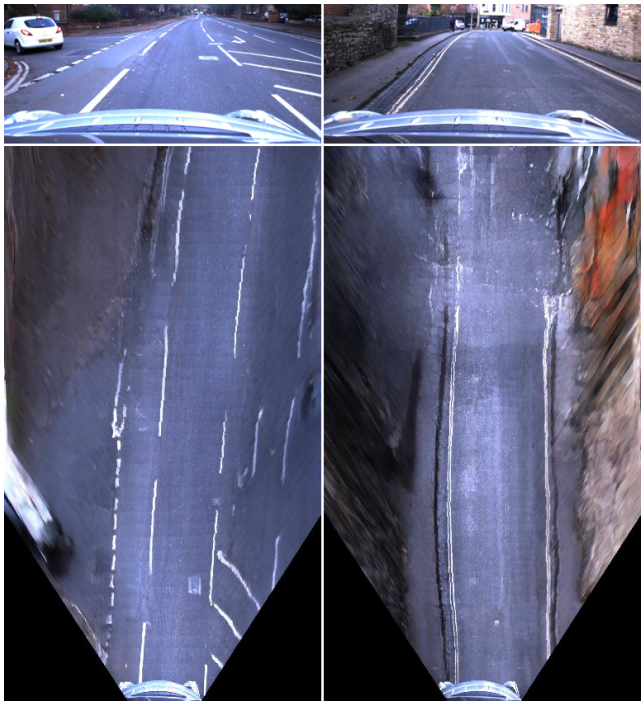


Fig. 6. Two cases in which the output of the network does not accurately depict the top-down view of the scene. In the *left* image, the road marking arrow is deformed, because we cannot employ a pixel-wise loss. In the *right* image, the road surface is not flat (sloping upwards), consequently the spatial transformer blocks attempt to map parts of the scene above the horizon, for which the features are not learned.

Consequently, our boosted IPM approach can have a significant impact on a wide range of applications in the context of autonomous driving including scene understanding, navigation, and planning.

References

[1] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 12, pp. 2481–2495, Dec 2017.

[2] L. Schneider, M. Cordts, T. Rehfeld, D. Pfeiffer, M. Enzweiler, U. Franke, M. Pollefeys, and S. Roth, "Semantic stixels: Depth is not enough," in *2016 IEEE Intelligent Vehicles Symposium (IV)*, June 2016, pp. 110–117.

[3] D. Neven, B. De Brabandere, S. Georgoulis, M. Proesmans, and L. Van Gool, "Towards end-to-end lane detection: An instance segmentation approach," *arXiv preprint arXiv:1802.05591*, 2018.

[4] W. Song, Y. Yang, M. Fu, Y. Li, and M. Wang, "Lane detection and classification for forward collision warning system based on stereo vision," *IEEE Sensors Journal*, vol. 18, no. 12, pp. 5151–5163, June 2018.

[5] B. Mathibela, P. Newman, and I. Posner, "Reading the road: Road marking classification and interpretation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 4, pp. 2072–2081, Aug 2015.

[6] A. L. Ballardini, D. Cattaneo, S. Fontana, and D. G. Sorrenti, "An online probabilistic road intersection detector," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, May 2017, pp. 239–246.

[7] S. Schulter, M. Zhai, N. Jacobs, and M. Chandraker, "Learning to look around objects for top-view representations of outdoor scenes," *arXiv preprint arXiv:1803.10870*, 2018.

[8] J. Dequaire, P. Ondrka, D. Rao, D. Wang, and I. Posner, "Deep tracking in the wild: End-to-end tracking using recurrent neural networks," *The International Journal of Robotics Research*, vol. 37, no. 4-5, pp. 492–512, 2018.

[9] N. Engel, S. Hoermann, P. Henzler, and K. Dietmayer, "Deep object tracking on dynamic occupancy grid maps using RNNs," *arXiv preprint arXiv:1805.08986*, 2018.

[10] N. Simond and M. Parent, "Obstacle detection from IPM and super-homography," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2007, pp. 4283–4288.

[11] N. Lee, W. Choi, P. Vernaza, C. B. Choy, P. H. S. Torr, and M. Chandraker, "DESIRE: Distant future prediction in dynamic scenes with interacting agents," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 2165–2174.

[12] A. Zyner, S. Worrall, and E. Nebot, "Naturalistic driver intention and path prediction using recurrent neural networks," *arXiv preprint arXiv:1807.09995*, 2018.

[13] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004.

[14] P. Isola, J. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 5967–5976.

[15] J. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017, pp. 2242–2251.

[16] X. Zhu, Z. Yin, J. Shi, H. Li, and D. Lin, "Generative adversarial frontal view to bird view synthesis," in *2018 International Conference on 3D Vision (3DV)*, Sep. 2018, pp. 454–463.

[17] M. Nieto, L. Salgado, F. Jaureguizar, and J. Cabrera, "Stabilization of inverse perspective mapping images based on robust vanishing point estimation," in *2007 IEEE Intelligent Vehicles Symposium*, June 2007, pp. 315–320.

[18] D. Zhang, B. Fang, W. Yang, X. Luo, and Y. Tang, "Robust inverse perspective mapping based on vanishing point," in *Proceedings 2014 IEEE International Conference on Security, Pattern Analysis, and Cybernetics (SPAC)*, Oct 2014, pp. 458–463.

[19] M. Bertozzi, A. Broggi, and A. Fascioli, "An extension to the inverse perspective mapping to handle non-flat roads," in *IEEE International Conference on Intelligent Vehicles. Proceedings of the 1998 IEEE International Conference on Intelligent Vehicles*, vol. 1, 1998.

[20] J. Jeong and A. Kim, "Adaptive inverse perspective mapping for lane map generation with SLAM," in *2016 13th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, Aug 2016, pp. 38–41.

[21] M. Oliveira, V. Santos, and A. D. Sappa, "Multimodal inverse perspective mapping," *Information Fusion*, vol. 24, pp. 108 – 121, 2015.

[22] C.-C. Lin and M.-S. Wang, "A vision based top-view transformation model for a vehicle parking assistant," *Sensors*, vol. 12, no. 4, pp. 4431–4446, 2012.

[23] P. Cerri and P. Grisleri, "Free space detection on highways using time correlation between stabilized sub-pixel precision ipm images," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, April 2005, pp. 2223–2228.

[24] J. M. M. García and N. Y. Ershadi, "A new strategy of detecting traffic information based on traffic camera : modified inverse perspective mapping," *Journal of Electrical Engineering, Technology and Interface Utilities*, vol. 10, no. 2, pp. 1101–1118, March 2017.

[25] S. Sengupta, P. Sturgess, L. Ladick, and P. H. S. Torr, "Automatic dense visual semantic mapping from street-level imagery," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2012, pp. 857–862.

[26] G. Máttyus, S. Wang, S. Fidler, and R. Urtasun, "HD maps: Fine-grained road segmentation by parsing ground and aerial images," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 3611–3619.

[27] M. Zhai, Z. Bessinger, S. Workman, and N. Jacobs, "Predicting ground-level scene layout from aerial imagery," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 4132–4140.

[28] K. Regmi and A. Borji, "Cross-view image synthesis using geometry-guided conditional GANs," *arXiv preprint arXiv:1808.05469*, 2018.

[29] T. Wang, M. Liu, J. Zhu, A. Tao, J. Kautz, and B. Catanzaro, "High-resolution image synthesis and semantic manipulation with conditional GANs," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2018, pp. 8798–8807.

[30] M. Jaderberg, K. Simonyan, A. Zisserman *et al.*, "Spatial transformer networks," in *Advances in neural information processing systems*, 2015, pp. 2017–2025.

[31] X. Yan, J. Yang, E. Yumer, Y. Guo, and H. Lee, "Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision," in *Advances in Neural Information Processing Systems*, 2016, pp. 1696–1704.

[32] T. Zhou, S. Tulsiani, W. Sun, J. Malik, and A. A. Efros, "View synthesis by appearance flow," in *European conference on computer vision*. Springer, 2016, pp. 286–301.

[33] D. J. Rezende, S. A. Eslami, S. Mohamed, P. Battaglia, M. Jaderberg, and N. Heess, "Unsupervised learning of 3D structure from images," in *Advances in Neural Information Processing Systems*, 2016, pp. 4996–5004.

[34] S. Azadi, D. Pathak, S. Ebrahimi, and T. Darrell, "Compositional GAN: Learning conditional image composition," *arXiv preprint arXiv:1807.07560*, 2018.

[35] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European Conference on Computer Vision*. Springer, 2016, pp. 694–711.

[36] A. Dosovitskiy and T. Brox, "Generating images with perceptual similarity metrics based on deep networks," in *Advances in Neural Information Processing Systems*, 2016, pp. 658–666.

[37] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[38] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[39] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016, pp. 770–778.

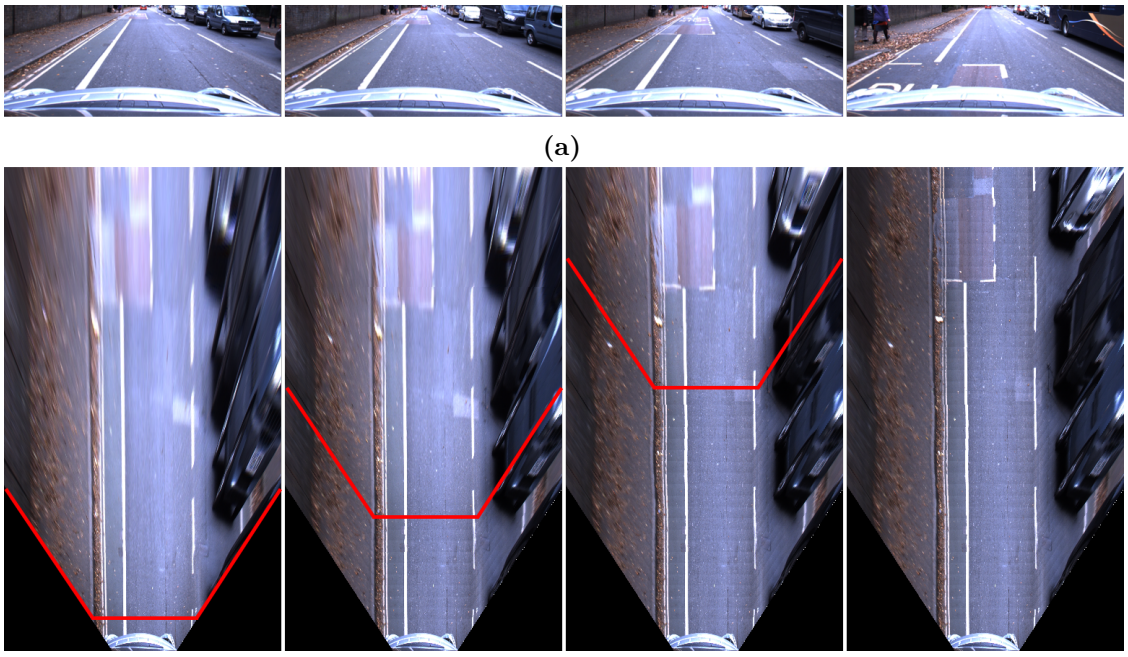
[40] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000 km: The Oxford RobotCar dataset," *The International Journal of Robotics Research*, vol. 36, no. 1, pp. 3–15, 2017.

[41] W. Churchill, "Experience based navigation: Theory, practice and implementation," Ph.D. dissertation, University of Oxford, Oxford, United Kingdom, 2012.

[42] C. Ledig, L. Theis, F. Huszr, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-realistic single image super-resolution using a generative adversarial network," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017, pp. 105–114.

[43] T. Bruls, W. Maddern, A. A. Morye, and P. Newman, "Mark yourself: Road marking segmentation via weakly-supervised annotations from multimodal data," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 1863–1870.

[44] L. Kunze, T. Bruls, T. Suleymanov, and P. Newman, "Reading between the lanes: Road layout reconstruction from partially segmented scenes," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, Nov 2018, pp. 401–408.



(a)

(b)

Figure 7.1: Generating a label for self-supervised boosted IPM training. From *left* to *right*, an increasing number of front-facing images, **(a)**, is mapped into IPM with the homography transformation. As this transformation is fairly accurate close to the vehicle, it is possible to generate a detailed bird's-eye view by stitching the nearby pixels into the label at the pixel locations according to the ego motion obtained from VO, **(b)**. The *red* line indicates the boundary up to which the label is stitched in every instance. The label becomes increasingly less distorted and better corresponding to the real world until the vehicle reaches the end of the field of view of the first image and the label is completed.

7.2 Further Details

Figure 7.1 visualizes the way a boosted IPM label is generated incrementally from VO and front-facing camera images, as explained in the publication (Section IV). Boosted IPM is learned in a self-supervised way and therefore does not require any human effort. The reader is referred to the video[2] accompanying our publication for a more detailed visualization of this process.

7.3 Further Results

This section presents further qualitative and quantitative results for several variants of boosted IPM and road marking segmentation therein.

[2] `https://www.youtube.com/watch?v=JLOAayZe1Do`

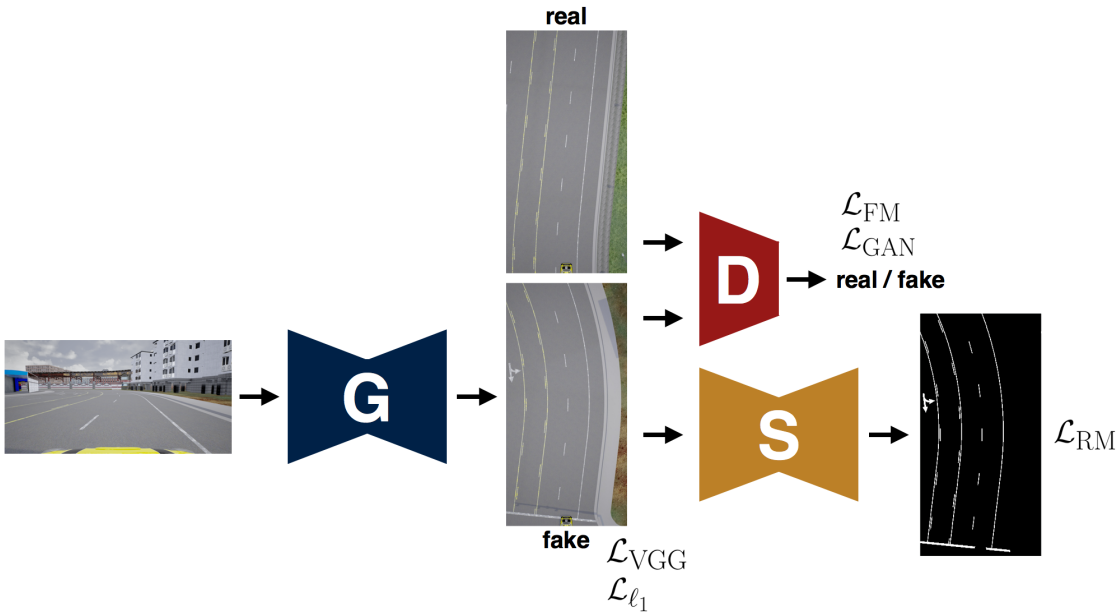


Figure 7.2: We evaluate three extensions of the standard boosted IPM framework, consisting of a generator G and a discriminator D with losses following [68]: (1) adding an ℓ_1-loss to boosted IPM, (2) adding a road marking task with a segmentation network S, and (3) a combination of the two aforementioned extensions. The CARLA driving simulator allows for pixel-wise evaluation as it provides a precise bird's-eye-view ground truth.

It is extremely difficult to acquire ground-truth labels in the real world due to the nature of the problem; therefore, we captured the required data in the CARLA driving simulator [60]. Since CARLA is a virtual environment, the viewpoint of the camera can be changed into a bird's-eye view to retrieve precise ground truth for training and testing the presented framework. We trained all variants on a sequence of 3000 images using the settings listed in the publication and tested on a different sequence of 1498 images.

7.3.1 Boosted IPM

The precise ground truth allows for evaluating three additional variants of boosted IPM, as visualized in Figure 7.2. Overall, we compare five different types of IPM:

1. Traditional homography-based IPM, estimated from manually-selected point correspondences.

2. Boosted IPM.

3. Boosted IPM (+ ℓ_1) with an additional ℓ_1-loss (with $\lambda_{\ell_1} = 10$ as in [110]).

4. Boosted IPM (+ RM) with a second output task to segment the road markings, which acts as an additional regularizer to the regularization achieved by the discriminator. This specifically enforces correct road markings in the boosted IPM, which are crucial for generating the scene graph. The segmentation network is equivalent to Chapter 5 and $\lambda_{\text{RM}} = 10$.

5. Boosted IPM (+ ℓ_1 + RM) with both the additional ℓ_1-loss and the road marking segmentation task.

Qualitative Evaluation

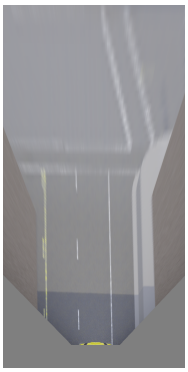
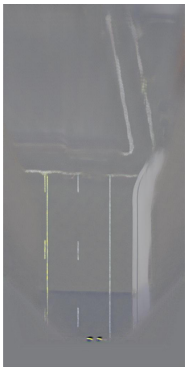
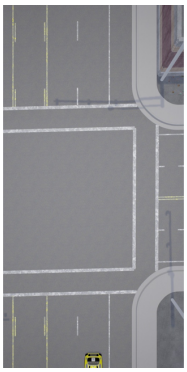
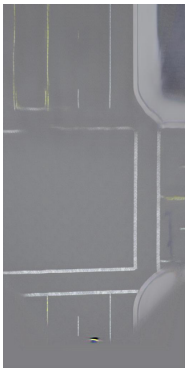
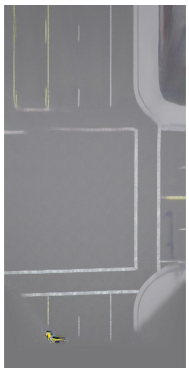
Table 7.1 presents qualitative results for boosted IPM in the CARLA environment. It includes results for (2) boosted IPM and (5) boosted IPM (+ ℓ_1 + RM) as the latter variant performs the best in the quantitative evaluation presented in Table 7.2. For all of the scenes, the learned IPM versions represent the geometry of the scene more accurately and contain sharper and more consistent road markings than (1) homography-based IPM. More specifically, the following observations are made for the various scenes:

- **Scene A**. The pedestrian crossing farther away is geometrically represented best by (5) boosted IPM (+ ℓ_1 + RM). This makes segmentation and registration easier and, in turn, allows the vehicle to adjust its behavior accordingly at an earlier time.

- **Scene B-D**. All but (5) boosted IPM (+ ℓ_1 + RM) fail to accurately generate the horizontal stop lines, which are critical for safe driving, at farther distances.

- **Scene E-F**. (1) Homography-based IPM blurs and stretches the road markings significantly due to the road cornering and pitching. The boosted IPM versions are not perfect but provide a more accurate representation of the actual road layout.

Table 7.1: A qualitative comparison of various types of IPM in the CARLA simulator.

Scene	Ground Truth	(1) Homography IPM	(2) Boosted IPM	(5) Boosted IPM (+ ℓ_1 + RM)
A 				
B 				
C 				
D 				

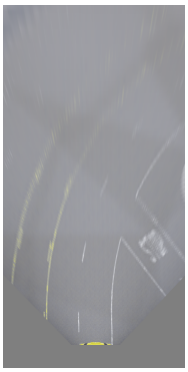
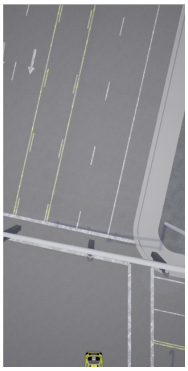
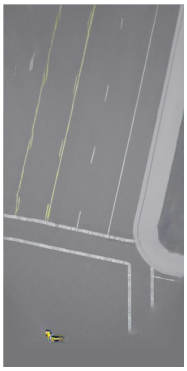
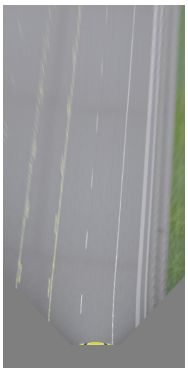
Scene	Ground Truth	(1) Homography IPM	(2) Boosted IPM	(5) Boosted IPM (+ ℓ_1 + RM)
E 				
F 				
G 				
H 				

Table 7.2: A quantitative comparison of various types of IPM in the CARLA simulator.

		FULL IMAGE			ROAD SURFACE	
		SSIM ↑	**PSNR ↑**	**VGG-loss ↓**	**SSIM ↑**	**PSNR ↑**
(1)	Homography IPM	0.8606	20.5530	0.5622	0.9266	23.7386
(2)	Boosted IPM	0.9096	26.1032	0.3669	0.9589	30.3472
(3)	Boosted IPM (+ ℓ_1)	0.9100	27.3554	0.3689	0.9583	31.9367
(4)	Boosted IPM (+ RM)	0.9099	26.1366	0.3633	0.9584	30.4551
(5)	Boosted IPM (+ ℓ_1 + RM)	**0.9134**	**27.8057**	**0.3632**	**0.9595**	**32.1687**

- **Scene G-H**. There is relatively little difference between the two boosted IPM variants. However, compared to (1) homography-based IPM, the lane geometries are easier to distinguish and more accurately resemble the ground truth. It is worth noting that (5) boosted IPM (+ ℓ_1 + RM) does extend the lane markings farther into the distance than (2) boosted IPM in scene H.

The reader is referred to the video[3] accompanying our publication for additional qualitative results on real-world data of the Oxford RobotCar dataset.

Quantitative Evaluation

Table 7.2 presents quantitative results for the various IPM types in the CARLA environment. Following [109], we evaluate three metrics: SSIM, PSNR, and a perceptual loss based on the VGG-network [68]. SSIM measures structural differences in the generated image, PSNR detects low-level differences, and the perceptual loss is calculated using learned features and designed to correlate well with human vision. We test on the full image as well as only on the road surface as the other parts of the IPM image are irrelevant for understanding the road layout.

The following observations are made from the results presented in the table:

- All boosted IPM versions perform better than (1) homography-based IPM across all metrics. Although homography-based IPM requires less computational effort, boosted IPM can also be generated online.

[3] `https://www.youtube.com/watch?v=JLOAayZe1Do`

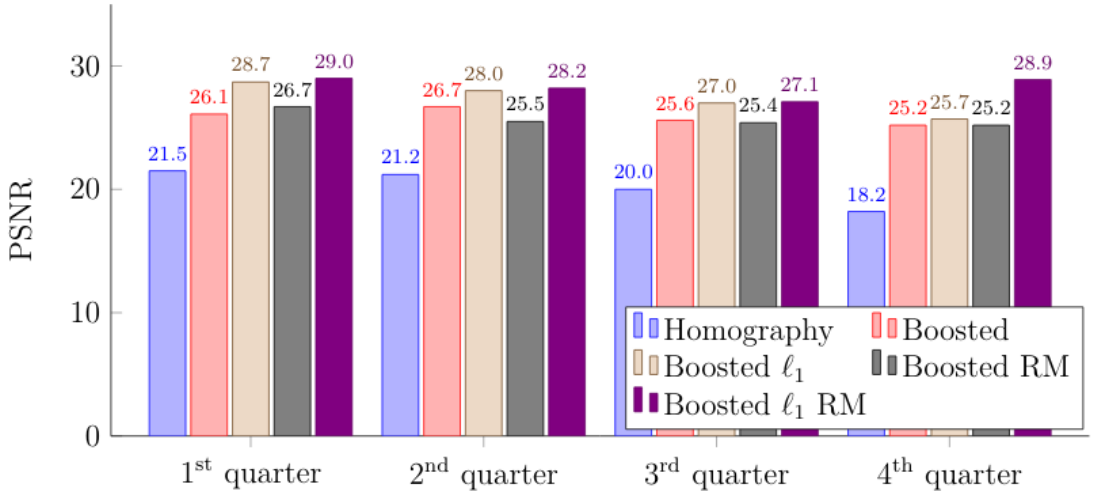


Figure 7.3: We evaluate the PSNR of the various types of IPM for four longitudinal quarters of the full image. The PSNR of (1) homography-based IPM decreases with distance, whereas it remains relatively constant for the boosted IPM variants.

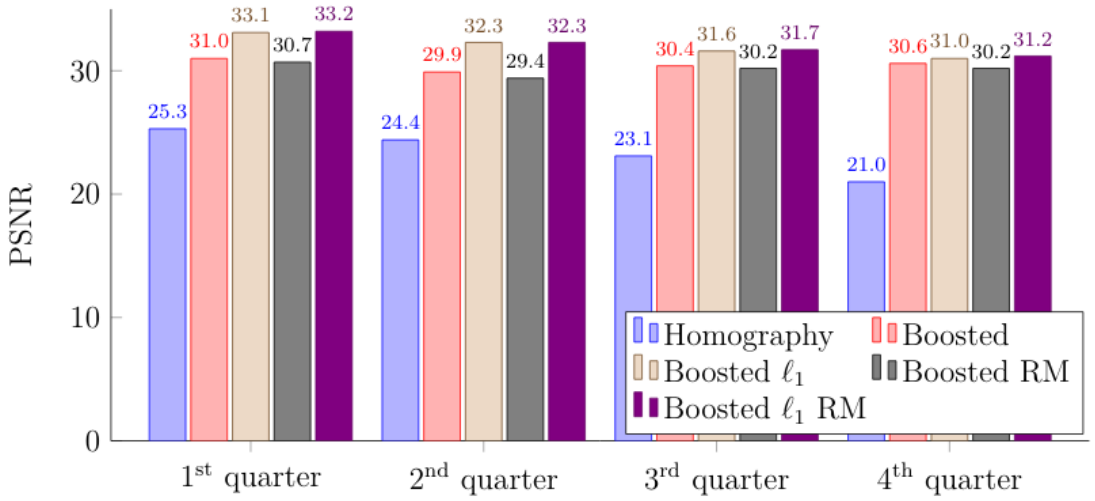


Figure 7.4: We evaluate the PSNR of the various types of IPM for four longitudinal quarters of the road surface. The PSNR of (1) homography-based IPM decreases with distance, whereas it remains relatively constant for the boosted IPM variants.

- (5) Boosted IPM (+ ℓ_1 + RM) outperforms all other methods. This is expected as the transformation to a perfect bird's-eye view can be seen as shape-altering and de-blurring.

- (3) Boosted IPM (+ ℓ_1) slightly outperforms (4) boosted IPM (+ RM), likely because we evaluate the full image/road surface and not just the road markings.

We also evaluate the differences between the IPM variants as a function of the distance for the full image and only the road surface in Figure 7.3 and Figure 7.4, respectively. We compare the PSNR for every quarter of the IPM image, where the first quarter is closest to the vehicle and the fourth quarter farthest away. It is clear that as the distance increases, the PSNR of (1) homography-based IPM starts to decrease, which is expected as the front-facing image contains fewer pixels in these regions resulting in blurred and stretched objects. The PSNR of the boosted IPM variants remains relatively constant across the quarters, and thus the most significant performance increase is achieved at farther distances. This aligns with the original motivation behind boosted IPM (and with [91]) to improve scene understanding at farther distances specifically.

7.3.2 Road Marking Segmentation in Bird's-Eye View

As demonstrated in the reproduced publication, the quality of the road marking segmentation impacts the quality of the generated scene graph directly. We have shown using real-world data that boosted IPM allows for more robust road marking segmentation (1) at greater distances and (2) in more detail, and (3) infers road markings occluded by dynamic objects. Here, we evaluate the benefits of boosted IPM for road marking segmentation qualitatively and quantitatively in the CARLA environment.

Following the methodology of [91] for high-way scenarios, we introduce another variant of boosted IPM, which we refer to as *boosted RL* (Road Layout). This variant generates a bird's-eye-view image that only represents the road layout (i.e. lane geometries and symbols) instead of the full image and is thus not tasked with modelling appearance, as visualized in Figure 7.5. This information is ultimately sufficient for generating the scene graphs studied in Chapter 4. We compare road marking segmentation in four variants of IPM split into two different cases:

- Training a road marking segmentation network equal to the one described in Chapter 5 on the generated (1) homography-based IPM, (2) boosted IPM, or (6) boosted RL (+ ℓ_1).

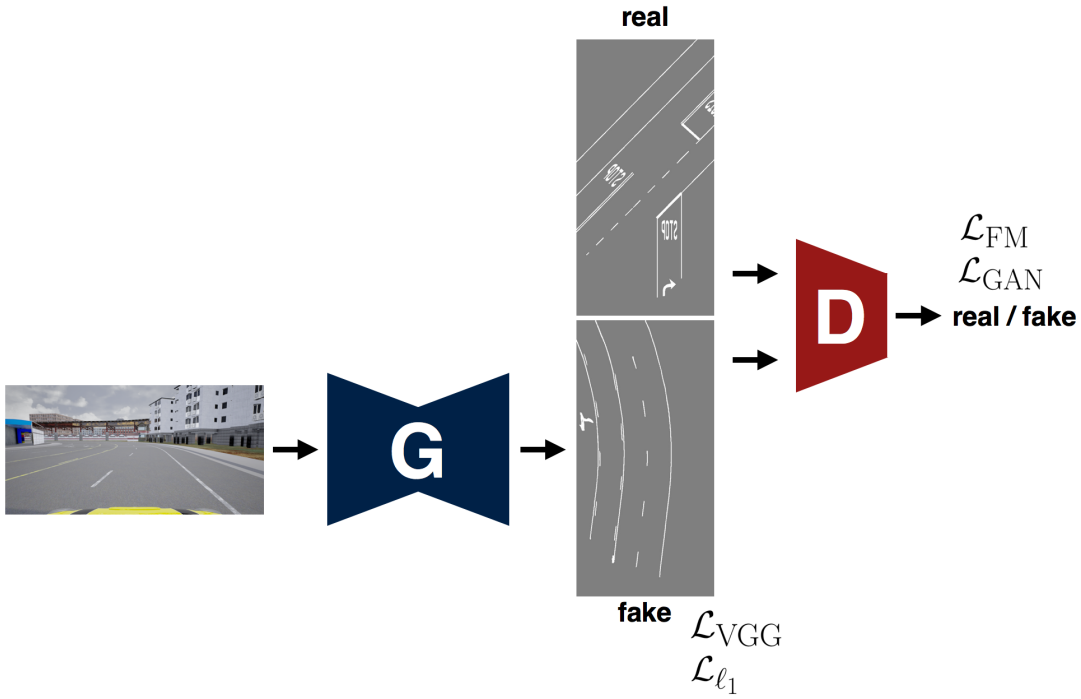


Figure 7.5: Instead of generating a full boosted IPM image, we adjust the framework to generate only the road layout (i.e. lane geometries and symbols) and refer to this variant as *boosted RL*. This allows the generator to focus on generating correct road markings instead of appearance, which in our case is sufficient for generating the scene graphs.

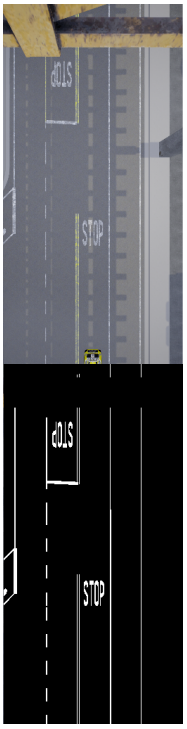
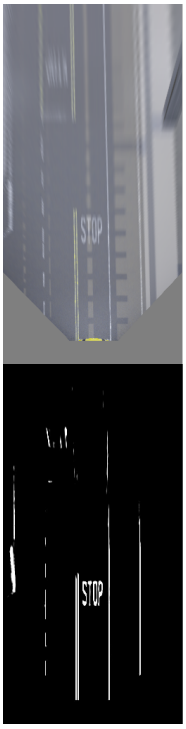
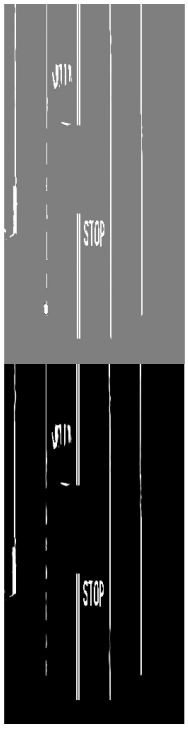
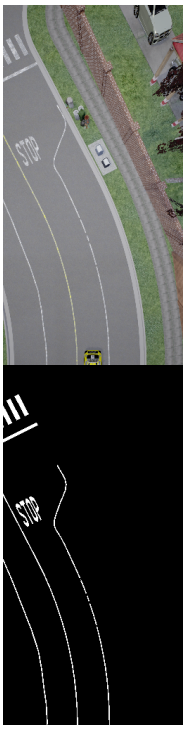
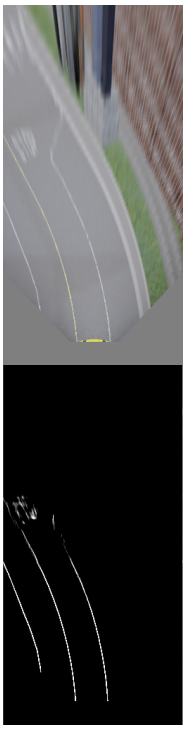
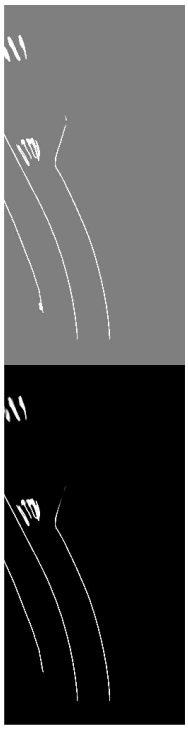
- Training a road marking segmentation network end-to-end in the boosted IPM framework (5) boosted IPM (+ ℓ_1 + RM).

For the first case, we only train the road marking segmentation network on the bottom half of the image (closest to the vehicle) but test on the full image. This prevents overfitting to the blurred and stretched road markings at a farther distance that do not accurately represent the true lane geometries. The network only achieves satisfactory performance during testing if the road markings are consistent in shape and size across the bird's-eye view, which is the distinguishing factor between homography-based and boosted IPM.

Qualitative Evaluation

Table 7.3 presents qualitative results for road marking segmentation in the various IPM variants in the CARLA environment. It is clear that the road markings become increasingly blurred and stretched as the distance increases in (1) homography-based

Table 7.3: A qualitative comparison of road marking segmentation in the various types of IPM in the CARLA simulator.

Scene	Ground Truth	(1) Homography IPM	(5) Boosted IPM (+ ℓ_1 + RM)	(6) Boosted RL (+ ℓ_1)
				
				

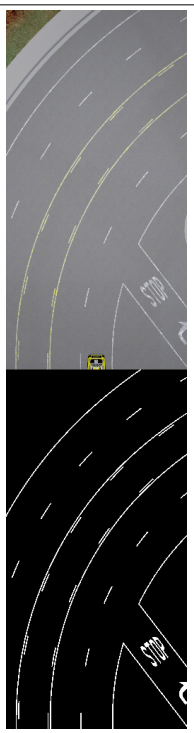
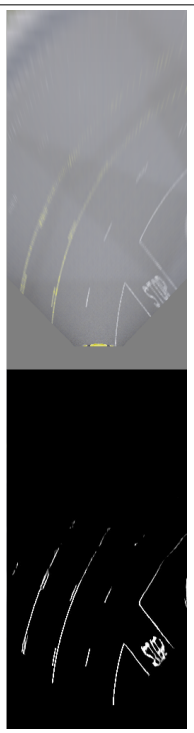
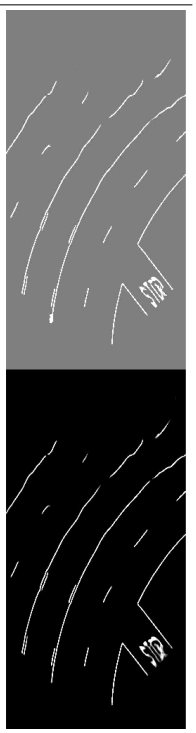
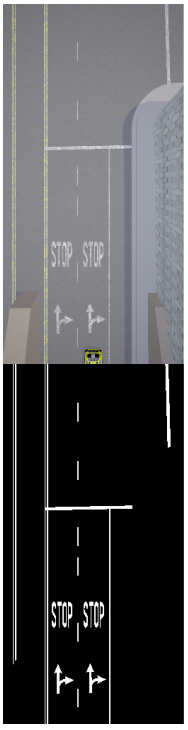
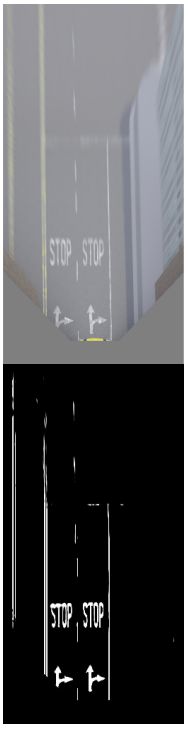
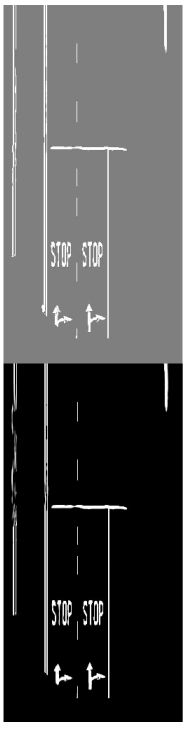
Scene	Ground Truth	(1) Homography IPM	(5) Boosted IPM (+ ℓ_1 + RM)	(6) Boosted RL (+ ℓ_1)
				
				

Table 7.4: A quantitative comparison of road marking segmentation in the various types of IPM in the CARLA simulator.

		Pre	Rec	$\mathbf{F_1}$	IoU
(1)	Homography IPM	64.15	43.07	50.63	35.69
(2)	Boosted IPM	58.49	56.92	57.19	44.16
(5)	Boosted IPM (+ ℓ_1 + RM)	**67.37**	54.92	59.35	46.43
(6)	Boosted RL (+ ℓ_1)	67.05	**57.30**	**61.00**	**48.26**

IPM; consequently, they are difficult to segment. In contrast, the boosted IPM variants provide sharper road markings that are better segmented and are therefore beneficial for generating scene graphs. (6) Boosted RL slightly outperforms (5) boosted IPM (+ ℓ_1 + RM), likely because the generator is forced to focus on generating only the road layout. Nevertheless, it remains challenging to generate symbols and letters accurately from the small number of pixels in the front-facing image with the current methods.

Quantitative Evaluation

Table 7.4 presents quantitative results for road marking segmentation for the various types of IPM. The following observations are made:

- All types of boosted IPM improve road marking segmentation substantially compared to (1) homography-based IPM. This is expected as boosted IPM provides sharper and more consistent road markings at farther distances.

- (4) Boosted IPM (+ ℓ_1 + RM) and (6) Boosted RL (+ ℓ_1) both perform better than (2) Boosted IPM. This indicates that it is beneficial to incorporate additional losses that directly enforce the correct generation of the road markings instead of only an image discriminator.

- (6) Boosted RL (+ ℓ_1) outperforms (4) Boosted IPM (+ ℓ_1 + RM), likely because the generator is able to focus entirely on generating correct road layouts instead of a full IPM image and its appearance.

7.4 Further Discussion

As illustrated in the publication, boosted IPM is able to predict the underlying road layout of areas that are occluded by either dynamic objects or overexposure. The main reason for this is that the stitching method potentially reveals previously occluded areas in the training label. We have demonstrated in the publication that the removal of dynamic objects is beneficial for interpreting the scene. However, not all cars are removed in boosted IPM, which is expected as some (parked) cars still appear in the stitched labels. An alternative approach for dealing with dynamic objects was recently published [32]. The bird's-eye view clearly indicates which areas cannot be observed from the front-facing images and are therefore predicted. This is vital information for safety-critical applications such as autonomous vehicles. Furthermore, we did not observe any noticeable problems related to severe illumination shifts within the labels. The most likely reason for this is that these labels occur relatively infrequently in the dataset. The CARLA simulator provides a reproducible environment for a thorough investigation of the influence of lighting, weather, and other traffic participants on boosted IPM. This is left for future research.

The presented framework currently has two main limitations. Firstly, the assumption of a planar road surface does not always hold. This gives rise to minor inaccuracies in the training labels or incorrect regions in the boosted IPM, as visualized in the publication. A straightforward way to improve the labels is to estimate the roll and pitch of the road surface from 3D information [111]. The network input can also be extended with such information, for instance, a height map of the road surface to improve the generated boosted IPM. Secondly, although boosted IPM generates more consistent road markings and thereby improves road marking segmentation, not all road layouts are generated accurately. The newly-introduced variant called boosted RL seems promising, but it might be beneficial to include domain knowledge regarding road construction to further restrict the generator towards realistic road markings and road layouts.

7.5 Conclusion

This chapter improved IPM, an important prerequisite for generating accurate scene graphs. Traditional homography-based IPM introduces (minor) inaccuracies that may lead to significant differences in the semantic interpretation of the scene. We have introduced boosted IPM, which is learned in an adversarial framework, to resolve these limitations.

Boosted IPM is generated by a new architecture called the Incremental Spatial Transformer GAN. It consists of a sequence of spatial transformers and ResNet blocks which incrementally map the front-facing image towards a bird's-eye view. We have introduced a way to generate training pairs automatically from front-facing images by leveraging VO and thereby facilitate self-supervised learning. Boosted IPM contains sharper road markings and more homogeneous illumination while dynamic objects are potentially removed from the scene, thus revealing the underlying road layout in an improved fashion.

We have demonstrated the positive effect of boosted IPM on the generation of scene graph qualitatively with real-world data. Since the performance of road marking segmentation improves in boosted IPM, it is beneficial for the interpretation of scenes and consequently can lead to safer planning and decision making. Furthermore, we evaluated various extensions of boosted IPM quantitatively in the CARLA simulator. This showed that including specific losses or tasks that focus on generating accurate road markings leads to improvements over the standard boosted IPM. More specifically, the newly-introduced boosted RL, which only generates the road layout, seems promising for future research.

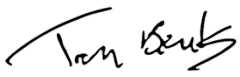
Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor **(only required where there isn't already a statement of contribution within the paper itself).**

Title of Paper	The Right (Angled) Perspective: Improving the Understanding of Road Scenes Using Boosted Inverse Perspective Mapping
Publication Status	Published
Publication Details	**T. Bruls**, H. Porav, L. Kunze, and P. Newman, "The right (angled) perspective: improving the understanding of road scenes using boosted inverse perspective mapping", in *Proceedings of the Intelligent Vehicles Symposium (IV)*, June 2019, pp. 302-309.

Student Confirmation

Student Name:	Tom Adriaan Hubert Bruls		
Contribution to the Paper	Contributions included: - Generating the initial ideas regarding boosted IPM, refining the ideas regarding the network architecture and the scene graph. - Developing the software for boosted IPM data processing and road marking detection. - Preparing and processing the IPM data. - Evaluating the experiments. - Performing the analysis. - Writing the paper, creating the figures and tables. - Presenting the work at the conference.		
Signature		Date	10-05-2020

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Professor Paul Newman			
Supervisor comments			
Signature		Date	11-05-2020

This completed form should be included in the thesis, at the end of the relevant chapter.

8

Summary and Future Directions

Contents

8.1 Summary 133
8.2 Broader Impact 136
8.3 Future Directions 137
8.3.1 Effective Representations for Road Scene Understanding 137
8.3.2 Scalable Learning for Road Scene Understanding 138
8.3.3 Appearance Invariance for Road Scene Understanding . 140
8.4 Closing Remarks 141

In this chapter, we review the work and research contributions presented in this thesis. We further present a discussion on the broader impact of the work and future directions of the three main themes.

8.1 Summary

This thesis devised effective representations for road scene understanding from in-situ perception, which can be employed directly for planning and decision making in various complex urban environments and under wide-ranging environmental conditions. Pixel-wise semantic segmentation is a common representation for road scene understanding in autonomous driving pipelines due to its versatility in describing a wide variety of scenes. However, it is limited in its direct usefulness

for decision making for several reasons, which we aimed to resolve in this thesis. In the process of achieving this, we have designed scalable learning techniques and improved robustness under adverse environmental conditions.

Chapter 3 extended semantic segmentation towards a wide range of adverse weather and lighting conditions. The segmentation performance decreases drastically in these cases because the input condition differs significantly from the overcast data on which the DNN was trained. This issue was resolved by implementing a framework of lightweight input adapters to map the different conditions to an appearance-invariant representation, which is the optimal input for the pretrained segmentation network. It was demonstrated that explicitly splitting these two tasks leads to better performance than training separate networks for each respective condition. Moreover, we employed image-to-image translation techniques to alter the appearance of an image while maintaining consistency with the semantic label, thereby generating vast quantities of training pairs for different conditions automatically.

Nevertheless, the resulting pixel-wise output representation (1) does not naturally support the high-level reasoning required for complex driving manoeuvres, (2) lacks a semantic understanding of the road rules (i.e road markings), and (3) is performed in the front-facing perspective, which does not align with the vehicle's action space. Chapter 4 resolved the first limitation by demonstrating a hybrid framework, which combines pixel-wise segmentations and object-centric perception to generate a graph-based scene description, the *scene graph*, that can be linked to the vehicle's action space and prior domain knowledge regarding road construction. This description requires various prerequisites such as curbs, a semantic understanding of the road markings, and an accurate IPM, which are challenging to obtain. We overcame some of these challenges in Chapter 5 - 7.

Chapter 5 presented a first step towards a semantic understanding of the road markings by training a DNN for binary segmentation. We limited the required labelling effort by leveraging LiDAR reflectance values and domain knowledge to generate vast quantities of training pairs automatically. Although these pairs might be approximations, we have demonstrated qualitatively that they are sufficient for

road marking segmentation if dropout is employed correctly. Furthermore, it was shown quantitatively that (pre)training on these pairs improves the performance in other domains where LiDAR and labelled data are unavailable. Even though this framework extends to different conditions, severe lens distortions due to raindrops significantly degrade the road marking segmentation performance. We demonstrated an image-to-image translation network that de-rains the images in order to resolve this issue. This method restored the segmentation performance and outperformed a network trained on rainy images and their corresponding labels.

However, the binary segmentation does not capture the road rules required for decision making, as conveyed by the semantic meaning of the road markings. Chapter 6 compared a model-driven and data-driven approach for road marking classification; both are implemented in a self-supervised way and employ the binary segmentation. The model-driven approach uses CORAL to fit linear models to the segmentation output and groups these models into semantic road marking instances by following road construction definitions. We showed qualitatively that this approach achieves an understanding of the road rules under various conditions. The data-driven approach employs image-to-image translation to synthesize a photo-realistic image for a predefined label, thereby generating vast quantities of the desired training pairs automatically. We showed quantitatively that these training pairs boost performance for rare road marking classes. Furthermore, we have introduced a new class-weighted cost function to retain performance in the presence of a large number of synthetic training pairs of a particular class.

Another essential prerequisite for generating the scene graph is accurate IPM. Traditional homography-based IPM deforms the shape of the road markings, which can lead to substantial errors in the semantic interpretation of scenes. Chapter 7 introduced a learned mapping from the front-facing image to a bird's-eye view, called *boosted IPM*. This view is generated by a new network architecture, called the Incremental Spatial Transformer GAN. The network is trained in a self-supervised way using VO and the sensor calibrations of the ego vehicle. Boosted IPM provides sharper road markings and a more homogeneous illumination while it allows for

reasoning about the road layout in occluded regions. We demonstrated qualitatively that this is beneficial for scene graph generation in real-world scenarios. We showed quantitatively in a physics-based simulator that our method outperforms homography-based IPM and is improved by adding losses and output tasks that act as additional regularizers.

8.2 Broader Impact

A long-standing debate that divides the AI community centres around whether we can solve any problem with more data and computational power or if there is still a need for exploiting human domain knowledge. Advancements such as AlphaGo (Zero) [112] seem to indicate the former; however, one should realise that the entire distribution of possibilities in simulated environments is narrow and therefore can be feasibly learned with finite resources.

In contrast, autonomous driving has an extremely long tail of different traffic situations and environmental conditions. One can quickly encounter scenarios that were not covered sufficiently during training. It seems inconvenient and most likely impossible (at least in the near future) to solve this problem by employing more data and computational power. The current solution encodes domain knowledge manually in order to work around these exceptions.

Although we have opted for a different approach in this thesis, HD-maps are a way of manually inserting domain knowledge into the system to make autonomous driving possible in geofenced areas. Another example of this is designing network architectures that explicitly learn or leverage mid-level representations, which humans envision to be valuable for learning the task of interest. This has proven to be beneficial in several recent works [113], [114].

We adopted the latter paradigm for autonomous driving in several chapters of this thesis by exploiting the fact that driving decisions are not influenced by environmental conditions (when accurate vehicle control is possible). This means that there must exist an appearance-invariant mid-level representation that encodes all necessary information. In Chapter 3, we introduced a framework that optimizes

explicitly for such a representation by "stripping" the appearance from the image. Similarly, we de-rained (i.e. stripped the appearance) images before segmenting the road markings in Section 5.4.

Lastly, we also demonstrated appearance-invariant and task-oriented (i.e. Boosted RL) representations in Chapter 7. These representations emerged from two similar realisations: (1) the scene graph is the same under all environmental conditions and (2) all of the required information for generating the scene graph is contained in the road markings and layout. It is, therefore, not necessary to model the image appearance accurately in the Boosted IPM. Some recent works [115], [116] have optimised this even further by representing road layouts with vectors.

Although devising the optimal representation for decision making in autonomous driving is still active research, we predict that the line of thinking demonstrated in this thesis will dominate the field for the foreseeable future.

8.3 Future Directions

This section discusses future directions for each of the underlying themes of this thesis.

8.3.1 Effective Representations for Road Scene Understanding

In Chapter 4, we demonstrated the scene graph, a hierarchical representation from in-situ perception that can be employed for decision making. However, the semantics of the scene (including the road markings) are currently not integrated into the scene graph as it is constructed from the binary segmentation. Integrating the semantics of the road markings is a crucial step towards real-world deployment as it will provide a planning algorithm with the information necessary to determine the appropriate driving behaviour.

This means that the following engineering and research steps are required in practice:

- The pixel-wise masks of the data-driven road marking classification system must be grouped into semantic instances, potentially by leveraging road construction definitions. It may also be interesting to investigate the level of detail required for accurate decision making. As discussed previously, it seems unnecessary to classify every pixel correctly. The class and approximate shape and location are most likely sufficient [117].

- These semantic instances could be combined with those from the model-driven approach and tracked over time.

- These instances and their semantic meaning need to be integrated into the scene graph. This will allow for a more detailed scene taxonomy and grammar, which increases the distinctiveness of particular configurations of road markings, similar to [79]. It will then be less challenging to determine the most likely scene graph from the probability distributions.

Another interesting research topic is the merging of the demonstrated representations built from in-situ perception with the ones stored in HD-maps (e.g. the ones provided by mapping services such as HERE and TomTom). This may seem merely an engineering task, but it is not trivial. Specific applications where maps are already used as priors to improve in-situ perception are object detection [118], behaviour prediction [115], [116], [119], and semantic segmentation [120], [121]. Similarly, maps could be used as prior information to determine the most likely scene graph.

8.3.2 Scalable Learning for Road Scene Understanding

Road scene understanding has become intrinsically linked with state-of-the-art computer vision solutions. Developments in data synthesis have opened up new possibilities for scalable learning during the course of this thesis. This is exemplified by our work on *road layout randomization* [20], which requires high-resolution image synthesis techniques that were infeasible when this work began. We expect that progress in learning techniques, especially in data generation, will lead to further improvements and new possibilities for road scene understanding tasks.

Several interesting questions have emerged recently around generative models for data synthesis. As we discussed in Chapter 5 and 6, there exists a complex interaction between the quality of the generated data and the network optimization. Inferring the optimal network architecture has received a substantial amount of interest [122]. A similar search to understand dataset optimality has recently sparked the interest of the computer vision community. This raises questions such as: "What kind of data does the learning process require?", "Does the optimal type of data change during the learning process?", and "How much data is actually required?" [123]–[125]. The ultimate goal is to generate the optimal dataset for the specific task as cheaply as possible. This has been explored as a toy problem by the authors of [126] but has not been applied yet for road scene understanding tasks. A better understanding of dataset optimality might break the performance plateau for road marking segmentation when trained on synthetic training data, which was hit in the reproduced publication in Section 6.2. In an ideal situation, we would continuously generate new data to bolster the performance of the current segmentation network without forgetting what was learned previously.

Another exciting direction is an extension of the cycle-consistency framework used in Chapter 3, which generated training data for multiple conditions. The current framework is only able to synthesize training pairs for conditions encountered in the captured data. However, not all real-world conditions will be present in the available datasets. This raises the following question: "Is it possible to synthesize unavailable conditions by combining various available conditions?". A concrete example of this is combining rainy daytime images and clear nighttime images to synthesize rainy nighttime images. Preliminary results for this are shown in Figure 8.1. This idea can be extended towards the continuous case, where the aim is to produce data along a weather condition and time-of-day (i.e. lighting) axis. In practice, this means disentangling the weather and lighting conditions from images in an unsupervised way, which is not trivial. Disentanglement is commonly performed for pose and appearance [127]–[129], but rarely for more complex road scenes [130]. One recently-published, concrete approach [131] disentangles lens occlusions (i.e.

(a) **(b)**

Figure 8.1: Preliminary results for combining separately-available weather and lighting conditions to synthesize new variations, which might not be available in the datasets. A mask of raindrops is added to four images of the same scene, **(b)**, after which we apply the appearance (i.e. lighting) of other images in the dataset, **(a)**. This process allows us to synthesize combinations of conditions such as rainy nighttime (*top left* and *bottom right* in **(b)**) that might not be available in the original dataset.

raindrops) from images. Alternatively, the problem can be considered as attribute control [132], [133], which generally requires labels. However, it remains challenging to disentangle complex weather and lighting conditions based on high-level image labels such as "rainy" and "nighttime".

8.3.3 Appearance Invariance for Road Scene Understanding

We showed in Chapter 3 and Chapter 7 that learning appearance-invariant representations and image restoration improves reasoning in occluded image regions. Nevertheless, for safety-critical applications such as autonomous vehicles, it is important to keep in mind that these outputs are predictions. They indicate what is likely to be found behind the occlusions but should be employed with caution and in combination with other sensor modalities. The image or particular image regions may be distorted to such an extent that it is impossible to reason about the scene accurately. Examples of these are dark unlit scenes during nighttime or images where large snowflakes/dirt cover most of the lens. It makes sense to incorporate some measure of uncertainty in the semantic segmentation for these cases in particular [134].

8.4 Closing Remarks

This thesis sought to contribute to road scene understanding for autonomous vehicles based on in-situ perception in complex urban environments. Our primary contributions are bridging the gap between the pixel-wise semantic scene segmentation and effective representations for decision making during real-world deployment, and we introduced scalable learning techniques in the process.

Learning generalized representations currently requires delicate supervision. Consequently, the role of self-supervised learning is becoming increasingly important in order to circumvent labelling costs. The presented approaches for that purpose can contribute to the development of safer transport in an autonomous world. Nevertheless, the task of road scene understanding from images remains challenging, and safe deployment anywhere and at any time will undoubtedly require more work, more thoughts, and more theses.

References

[1] U.S. Department of Transportation. (2016). "Automated driving systems 2.0: A vision for safety," [Online]. Available: `https://www.nhtsa.gov/sites/nhtsa.dot.gov/files/documents/13069a-ads2.0_090617_v9a_tag.pdf`.

[2] S. Ingle and M. Phute, "Tesla Autopilot: Semi autonomous driving, an uptick for future autonomy," *International Research Journal of Engineering and Technology (IRJET)*, vol. 3, no. 9, 2016.

[3] H. Xu, Y. Gao, F. Yu, and T. Darrell, "End-to-end learning of driving models from large-scale video datasets," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 3530–3538.

[4] M. Bansal, A. Krizhevsky, and A. Ogale, "ChauffeurNet: Learning to drive by imitating the best and synthesizing the worst," in *Proceedings of Conference on Robotics: Science and Systems (RSS)*, Jun. 2019.

[5] A. Bewley, J. Rigley, Y. Liu, *et al.*, "Learning to drive from simulation without real world labels," in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 4818–4824.

[6] J. S. Berrio, S. Worrall, M. Shan, and E. Nebot, "Long-term map maintenance pipeline for autonomous vehicles," *arXiv e-prints*, arXiv:2008.12449, Aug. 2020.

[7] D. Feng, C. Haase-Schütz, L. Rosenbaum, *et al.*, "Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges," *IEEE Transactions on Intelligent Transportation Systems (T-ITS)*, 2020.

[8] E. Yurtsever, J. Lambert, A. Carballo, and K. Takeda, "A survey of autonomous driving: Common practices and emerging technologies," *IEEE Access*, vol. 8, pp. 58 443–58 469, 2020.

[9] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia, "ICNet for real-time semantic segmentation on high-resolution images," in *Proceedings of the European Conference on Computer Vision (ECCV)*, Sep. 2018, pp. 418–434.

[10] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 6230–6239.

[11] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, Sep. 2018, pp. 833–851.

[12] G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognition Letters*, vol. 30, no. 2, pp. 88–97, 2009.

[13] G. Neuhold, T. Ollmann, S. R. Bulò, and P. Kontschieder, "The Mapillary Vistas Dataset for semantic understanding of street scenes," in *Proceedings of the International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 5000–5009.

[14] X. Huang, P. Wang, X. Cheng, *et al.*, "The ApolloScape open dataset for autonomous driving and its application," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 42, no. 10, pp. 2702–2719, 2020.

[15] S. Lee, J. Kim, J. S. Yoon, *et al.*, "VPGNet: Vanishing point guided network for lane and road marking detection and recognition," in *Proceedings of the International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 1965–1973.

[16] M. Müller, A. Dosovitskiy, B. Ghanem, and V. Koltun, "Driving policy transfer via modularity and abstraction," in *Proceedings of the Conference on Robot Learning (CORL)*, ser. Proceedings of Machine Learning Research, vol. 87, Oct. 2018, pp. 1–15.

[17] H. Porav, T. Bruls, and P. Newman, "Don't worry about the weather: Unsupervised condition-dependent domain adaptation," in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Oct. 2019, pp. 33–40.

[18] T. Bruls, W. Maddern, A. A. Morye, and P. Newman, "Mark yourself: Road marking segmentation via weakly-supervised annotations from multimodal data," in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 1863–1870.

[19] L. Kunze, T. Bruls, T. Suleymanov, and P. Newman, "Reading between the lanes: Road layout reconstruction from partially segmented scenes," in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Nov. 2018, pp. 401–408.

[20] T. Bruls, H. Porav, L. Kunze, and P. Newman, "Generating all the roads to Rome: Road layout randomization for improved road marking segmentation," in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Oct. 2019, pp. 831–838.

[21] ——, "The right (angled) perspective: Improving the understanding of road scenes using boosted inverse perspective mapping," in *Proceedings of the Intelligent Vehicles Symposium (IV)*, Jun. 2019, pp. 302–309.

[22] H. Porav, T. Bruls, and P. Newman, "I can see clearly now: Image restoration via de-raining," in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 7087–7093.

[23] P. Amayo, T. Bruls, and P. Newman, "Semantic classification of road markings from geometric primitives," in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Nov. 2018, pp. 387–393.

[24] F. Lateef and Y. Ruichek, "Survey on semantic segmentation using deep learning techniques," *Neurocomputing*, vol. 338, pp. 321–348, 2019.

[25] X. Liu, Z. Deng, and Y. Yang, "Recent progress in semantic image segmentation," *Artificial Intelligence Review*, vol. 52, no. 2, pp. 1089–1106, 2019.

[26] S. Sengupta, P. Sturgess, L. Ladický, and P. H. S. Torr, "Automatic dense visual semantic mapping from street-level imagery," in *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2012, pp. 857–862.

[27] C. Lu, M. J. G. van de Molengraft, and G. Dubbelman, "Monocular semantic occupancy grid mapping with convolutional variational encoder–decoder networks," *IEEE Robotics and Automation Letters (RA-L)*, vol. 4, no. 2, pp. 445–452, 2019.

[28] Ö. Erkent, C. Wolf, C. Laugier, D. S. Gonzalez, and V. R. Cano, "Semantic grid estimation with a hybrid bayesian and deep neural network approach," in *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2018, pp. 888–895.

[29] B. Pan, J. Sun, H. Y. T. Leung, A. Andonian, and B. Zhou, "Cross-view semantic segmentation for sensing surroundings," *IEEE Robotics and Automation Letters (RA-L)*, vol. 5, no. 3, pp. 4867–4873, 2020.

[30] T. Roddick and R. Cipolla, "Predicting semantic map representations from images using pyramid occupancy networks," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020, pp. 11 135–11 144.

[31] J. Philion and S. Fidler, "Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3D," in *Proceedings of the European Conference on Computer Vision (ECCV)*, Aug. 2020, pp. 194–210.

[32] L. Reiher, B. Lampe, and L. Eckstein, "A sim2real deep learning approach for the transformation of images from multiple vehicle-mounted cameras to a semantically segmented image in bird's eye view," in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Sep. 2020.

[33] A. Loukkal, Y. Grandvalet, T. Drummond, and Y. Li, "Driving among flatmobiles: Bird-eye-view occupancy grids from a monocular camera for holistic trajectory planning," in *Proceedings of the Winter Conference on Applications of Computer Vision (WACV)*, Jan. 2021, pp. 51–60.

[34] S. Schulter, M. Zhai, N. Jacobs, and M. Chandraker, "Learning to look around objects for top-view representations of outdoor scenes," in *Proceedings of the European Conference on Computer Vision (ECCV)*, Sep. 2018, pp. 815–831.

[35] K. Mani, S. Daga, S. Garg, *et al.*, "MonoLayout: Amodal scene layout from a single image," in *Proceedings of the Winter Conference on Applications of Computer Vision (WACV)*, Mar. 2020, pp. 1678–1686.

[36] K. Mani, N. Sai Shankar, K. Murthy Jatavallabhula, and K. Madhava Krishna, "Autolay: Benchmarking amodal layout estimation for autonomous driving," in *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2020, pp. 8184–8191.

[37] A. Ess, T. Müller, H. Grabner, and L. van Gool, "Segmentation-based urban traffic scene understanding," in *Proceedings of the British Machine Vision Conference (BMCV)*, Sep. 2009, pp. 84.1–84.11.

[38] A. Seff and J. Xiao, "Learning from maps: Visual common sense for autonomous driving," *arXiv e-prints*, arXiv:1611.08583, Nov. 2016.

[39] A. Geiger, M. Lauer, C. Wojek, C. Stiller, and R. Urtasun, "3D traffic scene understanding from movable platforms," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 36, no. 5, pp. 1012–1025, 2014.

[40] A. L. Ballardini, D. Cattaneo, S. Fontana, and D. G. Sorrenti, "An online probabilistic road intersection detector," in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, May 2017, pp. 239–246.

[41] J. Wang and J. Kim, "Semantic segmentation of urban scenes using spatial contexts," *IEEE Access*, vol. 8, pp. 55 254–55 268, 2020.

[42] G. Máttyus, S. Wang, S. Fidler, and R. Urtasun, "HD maps: Fine-grained road segmentation by parsing ground and aerial images," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 3611–3619.

[43] Z. Wang, B. Liu, S. Schulter, and M. Chandraker, "A parametric top-view representation of complex road scenes," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 10 317–10 325.

[44] B. Liu, B. Zhuang, S. Schulter, P. Ji, and M. Chandraker, "Understanding road layout from videos as a whole," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020, pp. 4413–4422.

[45] J.-B. Bordes, F. Davoine, P. Xu, and T. Denœux, "Evidential grammars: A compositional approach for scene understanding. application to multimodal street data," *Applied Soft Computing*, vol. 61, pp. 1173–1185, 2017.

[46] F. Dierkes, M. Raaijmakers, M. T. Schmidt, *et al.*, "Towards a multi-hypothesis road representation for automated driving," in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Sep. 2015, pp. 2497–2504.

[47] D. Töpfer, J. Spehr, J. Effertz, and C. Stiller, "Efficient road scene understanding for intelligent vehicles using compositional hierarchical models," *IEEE Transactions on Intelligent Transportation Systems (T-ITS)*, vol. 16, no. 1, pp. 441–451, Feb. 2015.

[48] S. K. Venkateshkumar, M. Sridhar, and P. Ott, "Latent hierarchical part based models for road scene understanding," in *Proceedings of the International Conference on Computer Vision Workshop (ICCVW)*, Dec. 2015, pp. 115–123.

[49] S. Mylavarapu, M. Sandhu, P. Vijayan, *et al.*, "Understanding dynamic scenes using graph convolution networks," *arXiv e-prints*, arXiv:2005.04437, May 2020.

[50] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *The International Journal of Robotics Research (IJRR)*, vol. 32, no. 11, pp. 1231–1237, 2013.

[51] H. Caesar, V. Bankiti, A. H. Lang, *et al.*, "nuScenes: A multimodal dataset for autonomous driving," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020, pp. 11 618–11 628.

[52] P. Sun, H. Kretzschmar, X. Dotiwalla, *et al.*, "Scalability in perception for autonomous driving: Waymo Open Dataset," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020, pp. 2443–2451.

[53] M. Cordts, M. Omran, S. Ramos, *et al.*, "The Cityscapes Dataset for semantic urban scene understanding," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 3213–3223.

[54] J. Geyer, Y. Kassahun, M. Mahmudi, *et al.*, "A2D2: Audi autonomous driving dataset," *arXiv e-prints*, arXiv:2004.06320, Apr. 2020.

[55] F. Yu, H. Chen, X. Wang, *et al.*, "BDD100K: A diverse driving dataset for heterogeneous multitask learning," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020, pp. 2633–2642.

[56] Z. Wang, B. Liu, S. Schulter, and M. Chandraker, "A dataset for high-level 3D scene understanding of complex road scenes in the top-view," in *Proceedings of the Conference on Computer Vision and Pattern Recognition Workshop (CVPRW)*, Jun. 2019.

[57] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 3234–3243.

[58] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for data: Ground truth from computer games," in *Proceedings of the European Conference on Computer Vision (ECCV)*, Oct. 2016, pp. 102–118.

[59] S. Shah, D. Dey, C. Lovett, and A. Kapoor, "AirSim: High-fidelity visual and physical simulation for autonomous vehicles," in *Field and Service Robotics (FSR)*, 2018, pp. 621–635.

[60] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An open urban driving simulator," in *Proceedings of the Conference on Robot Learning (CORL)*, ser. Proceedings of Machine Learning Research, vol. 78, Nov. 2017, pp. 1–16.

[61] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000 km: The Oxford Robotcar dataset," *The International Journal of Robotics Research (IJRR)*, vol. 36, no. 1, pp. 3–15, 2017.

[62] T. Scott, A. A. Morye, P. Piniés, *et al.*, "Choosing a time and place for calibration of LiDAR-camera systems," in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 4349–4356.

[63] C. Linegar, W. Churchill, and P. Newman, "Made to measure: Bespoke landmarks for 24-hour, all-weather localisation with a camera," in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, May 2016, pp. 787–794.

[64] G. Pascoe, "Robust lifelong visual navigation and mapping," Ph.D. dissertation, University of Oxford, 2017.

[65] W. S. Churchill, "Experience based navigation: Theory, practice and implementation," Ph.D. dissertation, University of Oxford, 2012.

[66] D. Barnes, W. Maddern, and I. Posner, "Find your own way: Weakly-supervised segmentation of path proposals for urban autonomy," in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, May 2017, pp. 203–210.

[67] W. Zhou, S. Worrall, A. Zyner, and E. Nebot, "Automated process for incorporating drivable path into real-time semantic segmentation," in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, May 2018, pp. 6039–6044.

[68] T. Wang, M. Liu, J. Zhu, *et al.*, "High-resolution image synthesis and semantic manipulation with conditional GANs," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2018, pp. 8798–8807.

[69] Z. Yang, Y. Chai, D. Anguelov, *et al.*, "SurfelGAN: Synthesizing realistic sensor data for autonomous driving," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020, pp. 11 115–11 124.

[70] M. Hahner, D. Dai, C. Sakaridis, J. Zaech, and L. van Gool, "Semantic understanding of foggy scenes with purely synthetic data," in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Nov. 2019, pp. 3675–3681.

[71] S. Azadi, M. Tschannen, E. Tzeng, *et al.*, "Semantic bottleneck scene generation," *arXiv e-prints*, arXiv:1911.11357, Nov. 2019.

[72] K. Kulkarni, T. Gokhale, R. Singh, P. Turaga, and A. Sankaranarayanan, "Halluci-Net: Scene completion by exploiting object co-occurrence relationships," *arXiv e-prints*, arXiv:2004.08614, Apr. 2020.

[73] E. Ntavelis, A. Romero, I. Kastanis, L. Van Gool, and R. Timofte, "SESAME: Semantic editing of scenes by adding, manipulating or erasing objects," in *Proceedings of the European Conference on Computer Vision (ECCV)*, Aug. 2020, pp. 394–411.

[74] W. Zhou, J. S. Berrio, S. Worrall, and E. Nebot, "Automated evaluation of semantic segmentation robustness for autonomous driving," *IEEE Transactions on Intelligent Transportation Systems (T-ITS)*, vol. 21, no. 5, pp. 1951–1963, 2020.

[75] D. Dai and L. van Gool, "Dark model adaptation: Semantic image segmentation from daytime to nighttime," in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Nov. 2018, pp. 3819–3824.

[76] L. Sun, K. Wang, K. Yang, and K. Xiang, "See clearer at night: Towards robust nighttime semantic segmentation through day-night image conversion," in *Artificial Intelligence and Machine Learning in Defense Applications*, vol. 11169, 2019, pp. 77–89.

[77] C. Sakaridis, D. Dai, and L. van Gool, "Guided curriculum model adaptation and uncertainty-aware evaluation for semantic nighttime image segmentation," in *Proceedings of the International Conference on Computer Vision (ICCV)*, Oct. 2019, pp. 7373–7382.

[78] T. Suleymanov, P. Amayo, and P. Newman, "Inferring road boundaries through and despite traffic," in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Nov. 2018, pp. 409–416.

[79] B. Mathibela, P. Newman, and I. Posner, "Reading the road: Road marking classification and interpretation," *IEEE Transactions on Intelligent Transportation Systems (T-ITS)*, vol. 16, no. 4, pp. 2072–2081, 2015.

[80] D. Wang, C. Devin, Q.-Z. Cai, P. Krähenbühl, and T. Darrell, "Monocular plan view networks for autonomous driving," *arXiv e-prints*, arXiv:1905.06937, May 2019.

[81] X. Li, Q. Xue, J. Zhao, and D. Wang, "Causal reasoning in multi-object interaction on the traffic scene: Occlusion-aware prediction of visibility fluent," *IEEE Access*, vol. 8, pp. 80 527–80 535, 2020.

[82] M. Gadd, D. de Martini, L. Marchegiani, P. Newman, and L. Kunze, "Sense-Assess-eXplain (SAX): Building trust in autonomous vehicles in challenging real-world driving scenarios," in *Proceedings of the Intelligent Vehicles Symposium (IV)*, Oct. 2020, pp. 150–155.

[83] L. Liu, H. Ma, S. Chen, *et al.*, "Image-translation-based road marking extraction from mobile laser point clouds," *IEEE Access*, vol. 8, pp. 64 297–64 309, 2020.

[84] L. Ma, Y. Li, J. Li, *et al.*, "Capsule-based networks for road marking extraction and classification from mobile LiDAR point clouds," *IEEE Transactions on Intelligent Transportation Systems (T-ITS)*, 2020.

[85] T. Wu and A. Ranganathan, "A practical system for road marking detection and recognition," in *Proceedings of the Intelligent Vehicles Symposium (IV)*, Jun. 2012, pp. 25–30.

[86] T. M. Hoang, P. H. Nguyen, N. Q. Truong, Y. W. Lee, and K. R. Park, "Deep RetinaNet-based detection and classification of road markings by visible light camera sensors," *Sensors*, vol. 19, no. 2, 2019.

[87] X.-Y. Ye, D.-S. Hong, H.-H. Chen, P.-Y. Hsiao, and L.-C. Fu, "A two-stage real-time YOLOv2-based road marking detector with lightweight spatial transformation-invariant classification," *Image and Vision Computing*, vol. 102, p. 103 978, 2020.

[88] Y. Hou, Z. Ma, C. Liu, T.-W. Hui, and C. C. Loy, "Inter-region affinity distillation for road marking segmentation," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020, pp. 12 483–12 492.

[89] D. Zhang, B. Fang, W. Yang, X. Luo, and Y. Tang, "Robust inverse perspective mapping based on vanishing point," in *Proceedings of the International Conference on Security, Pattern Analysis, and Cybernetics (SPAC)*, Oct. 2014, pp. 458–463.

[90] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004.

[91] Z. Yu, X. Ren, Y. Huang, W. Tian, and J. Zhao, "Detecting lane and road markings at a distance with perspective transformer layers," in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Sep. 2020.

[92] T. Veit, J. Tarel, P. Nicolle, and P. Charbonnier, "Evaluation of road marking feature extraction," in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Oct. 2008, pp. 174–181.

[93] S. Chen, Z. Zhang, R. Zhong, *et al.*, "A dense feature pyramid network-based deep learning model for road marking instance segmentation using mls point clouds," *IEEE Transactions on Geoscience and Remote Sensing (TGRS)*, vol. 59, no. 1, pp. 784–800, 2021.

[94] J. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the International Conference on Computer Vision (ICCV)*, Oct. 2017, pp. 2242–2251.

[95] X. Chen, S. Wang, B. Fu, M. Long, and J. Wang, "Catastrophic forgetting meets negative transfer: Batch spectral shrinkage for safe transfer learning," in *Advances in Neural Information Processing Systems*, vol. 32, 2019, pp. 1908–1918.

[96] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, vol. 39, no. 12, pp. 2481–2495, 2017.

[97] C. Xie, M. Tan, B. Gong, *et al.*, "Adversarial examples improve image recognition," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020, pp. 816–825.

[98] T. Chen, Z. Chen, Q. Shi, and X. Huang, "Road marking detection and classification using machine learning algorithms," in *Proceedings of the Intelligent Vehicles Symposium (IV)*, Jun. 2015, pp. 617–621.

[99] M. Uřičář, J. Uličný, G. Sistu, *et al.*, "Desoiling dataset: Restoring soiled areas on automotive fisheye cameras," in *Proceedings of the International Conference on Computer Vision Workshop (ICCVW)*, Oct. 2019, pp. 4273–4279.

[100] R. Shetty, B. Schiele, and M. Fritz, "Not using the car to see the sidewalk — quantifying and controlling the effects of context in classification and segmentation," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 8210–8218.

[101] P. Amayo, P. Piniés, L. M. Paz, and P. Newman, "Geometric multi-model fitting with a convex relaxation algorithm," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2018, pp. 8138–8146.

[102] E. D. Cubuk, B. Zoph, D. Mané, V. Vasudevan, and Q. V. Le, "AutoAugment: Learning augmentation strategies from data," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 113–123.

[103] R. Krajewski, T. Moers, and L. Eckstein, "VeGAN: Using GANs for augmentation in latent space to improve the semantic segmentation of vehicles in images from an aerial perspective," in *Proceedings of the Winter Conference on Applications of Computer Vision (WACV)*, Jan. 2019, pp. 1440–1448.

[104] T. Park, M. Liu, T. Wang, and J. Zhu, "Semantic image synthesis with spatially-adaptive normalization," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 2332–2341.

[105] J. Devaranjan, A. Kar, and S. Fidler, "Meta-Sim2: Unsupervised learning of scene structure for synthetic data generation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, Aug. 2020, pp. 715–733.

[106] J. Tobin, R. Fong, A. Ray, *et al.*, "Domain randomization for transferring deep neural networks from simulation to the real world," in *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2017, pp. 23–30.

[107] M. Oliveira, V. Santos, and A. D. Sappa, "Multimodal inverse perspective mapping," *Information Fusion*, vol. 24, pp. 108–121, 2015.

[108] J.-K. Lee, Y.-K. Baik, H. Cho, and S. Yoo, "Online extrinsic camera calibration for temporally consistent IPM using lane boundary observations with a lane width prior," *arXiv e-prints*, arXiv:2008.03722, Aug. 2020.

[109] X. Zhu, Z. Yin, J. Shi, H. Li, and D. Lin, "Generative adversarial frontal view to bird view synthesis," in *Proceedings of the International Conference on 3D Vision (3DV)*, Sep. 2018, pp. 454–463.

[110] J.-Y. Zhu, R. Zhang, D. Pathak, *et al.*, "Toward multimodal image-to-image translation," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 465–476.

[111] S. Boschenriedter, P. Hossbach, C. Linnhoff, S. Luthardt, and S. Wu, "Multi-session visual roadway mapping," in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Nov. 2018, pp. 394–400.

[112] D. Silver, J. Schrittwieser, K. Simonyan, *et al.*, "Mastering the game of Go without human knowledge," *Nature*, vol. 550, no. 7676, pp. 354–359, 2017.

[113] A. Sax, J. O. Zhang, B. Emi, *et al.*, "Learning to navigate using mid-level visual priors," *arXiv e-prints*, arXiv:1912.11121, Dec. 2019.

[114] B. Zhou, P. Krähenbühl, and V. Koltun, "Does computer vision matter for action?" *arXiv e-prints*, arXiv:1905.12887, May 2019.

[115] J. Gao, C. Sun, H. Zhao, *et al.*, "VectorNet: Encoding HD maps and agent dynamics from vectorized representation," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020, pp. 11 522–11 530.

[116] M. Liang, B. Yang, R. Hu, *et al.*, "Learning lane graph representations for motion forecasting," in *Proceedings of the European Conference on Computer Vision (ECCV)*, Aug. 2020, pp. 541–556.

[117] A. Behl, K. Chitta, A. Prakash, E. Ohn-Bar, and A. Geiger, "Label efficient visual abstractions for autonomous driving," *arXiv e-prints*, arXiv:2005.10091, May 2020.

[118] B. Yang, M. Liang, and R. Urtasun, "HDNET: Exploiting HD maps for 3D object detection," in *Proceedings of the Conference on Robot Learning (CORL)*, ser. Proceedings of Machine Learning Research, vol. 87, Oct. 2018, pp. 146–155.

[119] S. Casas, C. Gulino, S. Suo, and R. Urtasun, "The importance of prior knowledge in precise multimodal prediction," *arXiv e-prints*, arXiv:2006.02636, Jun. 2020.

[120] J. Wang and J. Kim, "Semantic segmentation of urban scenes with a location prior map using LiDAR measurements," in *Proceedings of the International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2017, pp. 661–666.

[121] A. Loukkal, V. Fremont, Y. Grandvalet, and Y. Li, "Improving semantic segmentation in urban scenes with a cartographic information," in *Proceedings of the International Conference on Control, Automation, Robotics and Vision (ICARCV)*, Nov. 2018, pp. 400–406.

[122] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2018, pp. 8697–8710.

[123] V. Besnier, H. Jain, A. Bursuc, M. Cord, and P. Pérez, "This dataset does not exist: Training models from generated images," in *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2020.

[124] S. Cheng, Z. Leng, E. Dogus Cubuk, *et al.*, "Improving 3D object detection through progressive population based augmentation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, Aug. 2020, pp. 279–294.

[125] W. Li, Z. Wang, Y. Yue, *et al.*, "Semi-supervised learning using adversarial training with good and bad samples," *Machine Vision and Applications*, vol. 31, no. 6, p. 49, 2020.

[126] F. P. Such, A. Rawal, J. Lehman, K. Stanley, and J. Clune, "Generative teaching networks: Accelerating neural architecture search by learning to generate synthetic training data," in *Proceedings of the International Conference on Machine Learning (ICML)*, ser. Proceedings of Machine Learning Research, vol. 119, Jul. 2020, pp. 9206–9216.

[127] X. Huang, M.-Y. Liu, S. Belongie, and J. Kautz, "Multimodal unsupervised image-to-image translation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, Sep. 2018, pp. 179–196.

[128] Y. Li, C. Twigg, Y. Ye, L. Tao, and X. Wang, "Disentangling pose from appearance in monochrome hand images," in *Proceedings of the International Conference on Computer Vision Workshop (ICCVW)*, Oct. 2019, pp. 2846–2855.

[129] L. Yang and A. Yao, "Disentangling latent hands for image synthesis and pose estimation," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 9869–9878.

[130] F. Xiao, H. Liu, and Y. J. Lee, "Identity from here, pose from there: Self-supervised disentanglement and generation of objects using unlabeled videos," in *Proceedings of the International Conference on Computer Vision (ICCV)*, Oct. 2019, pp. 7012–7021.

[131] F. Pizzati, P. Cerri, and R. de Charette, "Model-based disentanglement of lens occlusions," *arXiv e-prints*, arXiv:2004.01071, Apr. 2020.

[132] M. Liu, Y. Ding, M. Xia, *et al.*, "STGAN: A unified selective transfer network for arbitrary image attribute editing," in *Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2019, pp. 3668–3677.

[133] A. Mukherjee, A. Joshi, S. Sarkar, and C. Hegde, "Attribute-controlled traffic data augmentation using conditional generative models," in *Proceedings of the Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Jun. 2019, pp. 83–87.

[134] J. Tian, W. Cheung, N. Glaser, Y.-C. Liu, and Z. Kira, "UNO: Uncertainty-aware Noisy-Or multimodal fusion for unanticipated input degradation," in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, May 2020, pp. 5716–5723.

Appendices

A

Reading between the Lanes: Road Layout Reconstruction from Partially Segmented Scenes

This appendix contains a reproduction of the following publication:

> [19] L. Kunze, **T. Bruls**, T. Suleymanov, and P. Newman, "Reading between the lanes: Road layout reconstruction from partially segmented scenes", in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Nov. 2018, pp. 401-408.

Reading between the Lanes: Road Layout Reconstruction from Partially Segmented Scenes

Lars Kunze, Tom Bruls, Tarlan Suleymanov, and Paul Newman

***Abstract*— Autonomous vehicles require an accurate and adequate representation of their environment for decision making and planning in real-world driving scenarios. While deep learning methods have come a long way providing accurate semantic segmentation of scenes, they are still limited to pixel-wise outputs and do not naturally support high-level reasoning and planning methods that are required for complex road manoeuvres. In contrast, we introduce a hierarchical, graph-based representation, called *scene graph*, which is reconstructed from a partial, pixel-wise segmentation of an image, and which can be linked to domain knowledge and AI reasoning techniques.**

In this work, we use an adapted version of the Earley parser and a learnt probabilistic grammar to generate scene graphs from a set of segmented entities. Scene graphs model the structure of the road using an abstract, logical representation which allows us to link them with background knowledge. As a proof-of-concept we demonstrate how parts of a parsed scene can be inferred and classified beyond labelled examples by using domain knowledge specified in the Highway Code. By generating an interpretable representation of road scenes and linking it to background knowledge, we believe that this approach provides a vital step towards explainable and auditable models for planning and decision making in the context of autonomous driving.

I. Introduction

Autonomous vehicles need to perceive their surroundings accurately for safe decision making and navigation in complex urban environments. These highly-structured environments can be described by hierarchical graphs containing semantic and spatial constraints. Such graphical representations can be employed for (cost-based) planning, inferring object classes, or reasoning about missing or occluded parts. More importantly, they provide a way to explain the behaviour and decision making of the vehicle which is paramount for real-world deployment and adoption. In this paper, we introduce such a representation, which is generated from partially segmented scenes and allows us to reason about the environment.

Recently, deep semantic segmentation networks have achieved impressive results for pixel-wise scene understanding of images [1], [2]. However, these methods suffer from interpretation and debugging difficulties and often fail to include prior information or dependencies/constraints (in the output space). More importantly, they do not naturally support high-level reasoning which is required for planning and navigation.

Authors are from the Oxford Robotics Institute, Dept. Engineering Science, University of Oxford, UK. {lars, tombruls, tarlan, pnewman}@robots.ox.ac.uk

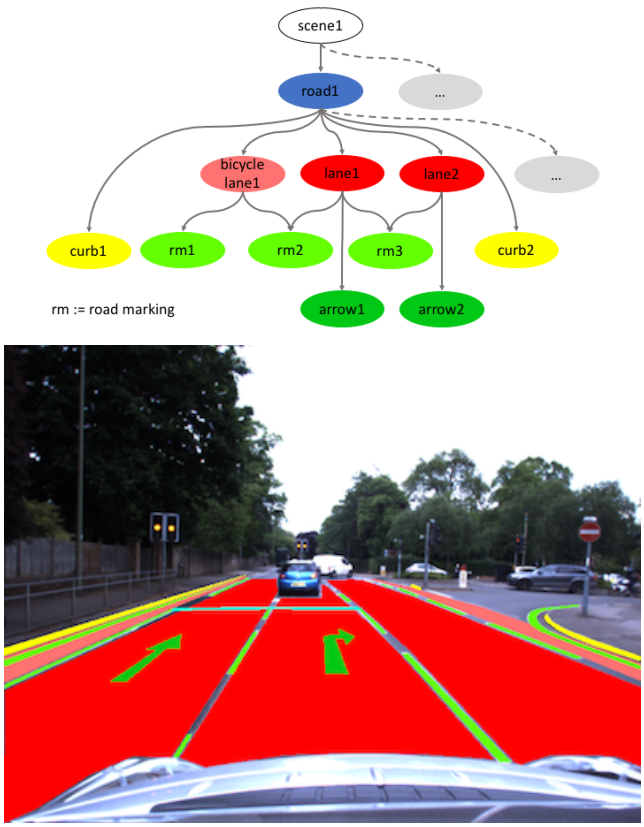


Fig. 1. Hierarchical scene graph representation (top) that was reconstructed from a partially segmented image (bottom). In this work we present a probabilistic scene parser that reconstructs the layout of road scenes from partial segmentations of road markings and curbs.

In contrast, all important (static or dynamic) objects influencing the decision making are detected separately in the mediated approach [3], [4]. This produces a world representation which can be employed more directly for planning and navigation. Interestingly, most approaches focus on detecting a single type of object or perform detection of several types of objects independently. Thereby they neglect that urban traffic scenes are highly structured and that there exist spatial and semantic constraints between objects, since these scenes are built and function according to specified rules.

Therefore, we introduce *scene graphs*, a hierarchical, graph-based representation, to model road layouts (i.e. lane geometries). Fig. 1 shows an example scene graph for a segmented road scene. We focus on the reconstruction of scene graphs from partial, pixel-wise segmentation. In par-

ticular, we consider segmented entities of road markings and curbs to reconstruct the semantic structure of road scenes. The road layout is reconstructed from these entities using both a learnt probabilistic context-free grammar and a learnt spatial, relational model. A road layout is chosen from a set of competing hypotheses by estimating the maximum a posteriori probability (MAP) of each model. Furthermore, we show that scene graphs can be refined by linking them with domain knowledge about the road construction, e.g. from the Highway Code.

In this paper, we make the following contributions:

- we introduce *scene graph*, a formal logic-based description of road scenes using a graph-based representation;
- we present an approach based on dynamic programming for parsing road scenes and reconstructing scene graphs from partial, segmentations and a learnt probabilistic grammar; and
- we demonstrate how scene graphs can be further refined and used for reasoning when linked to domain knowledge.

The remainder of the paper is structured as follows. We first discuss related work in Sec. II. In Sec. III, we provide and overview of the approach and explain how scene graphs are generated from both object segmentations and learnt prior models. In Sec. IV, we explain how scenes are partially segmented using deep networks for road markings and curbs. In Sec. V we explain how we represent a scene, learn both a probabilistic context-free grammar and a spatial relational model to describe scenes, and how scenes are parsed and interpreted using an adapted version of the probabilistic Earley parser. In Sec. VI, we showcase and discuss several examples of scene graphs and explain how they can be further refined. Lastly, we discuss possible application in Sec. VII before we conclude in Sec. VIII.

II. Related Work

In this section, we review different approaches for scene understanding in the context of autonomous vehicles. We mainly focus on graph based methods, since these are closest to the scene graph.

1) Graph-based Approaches: Representing the contents of scenes using graph-based approaches is not novel. In the context of urban traffic scenes, however, there exist only a few papers that take the spatial and semantic constraints into account by introducing graphs.

In [5], different sensor modalities and hierarchical graphs containing relational knowledge are fused to model traffic scenes. The output is still a pixel-wise segmented image not directly employable for automated driving.

Several other papers implement more high-level reasoning to infer the lane geometries. The authors of [6] introduce a theoretical, hierarchical framework including uncertainties to reason about multiple hypotheses for the lane geometry. Similar methods that work on real-world data are introduced in [7], [8]. From linear patches of lane markings a graph is built including their spatial relationship represented by continuous distributions and non-parametric belief propagation is used to infer the different lanes in the scene. However, these methods are not guaranteed to work in urban environments.

In [9], the lane separators are modelled as latent variables without linear constraints so that the framework becomes applicable to more complex scenes. By encoding geometric relationships at different levels (i.e. lane markings, lane separators, lanes, and road), the authors show that they improve inference of the lane geometries even in case of many false detections at the root nodes. This work is similar to our approach as we also represent the geometric relationships of different entities according to the hierarchy.

The driving rules of a traffic scene are given by the type of road markings that often appear in similar configurations. Therefore, [10] connects them as a graph and optimises a CRF with handcrafted spatial features of the road markings to predict their class. Similarly, we learn a distribution of geometric and relation features to predict and evaluate the type and the role of an entity within the hierarchy.

Work by [11] is most similar to our approach. In their work, they learn a probabilistic grammar based on a set of features and use a dynamic programming approach to generate a scene graphs which describe the furniture layout of synthetic indoor scenes. Whereas their approach considers full object knowledge from CAD models, our approach reconstructs scenes from partial observations of real-world environments.

2) Mediated Approaches: Proposed solutions differ widely in terms of the objects that are taken into account, used sensors, required computation time, usage of prior information, and abstraction level of the output. In general, our approach is flexible to consider different kinds of information from various resources. In this work, we consider segments of road markings and curbs as input.

In [12] a coarse road geometry/scene analysis is estimated from the acquired semantic segmentation. This framework is significantly extended in [3] where the precise intersection geometry is inferred from vanishing points, semantic labels, and tracklets of traffic participants. However, these methods cannot be used for navigation directly as they do not map to precise lane geometries and do not include the road rules. The former is solved in [13] by looking more closely into the tracks of the surrounding vehicles. Our also approach models the geometry of high-level concepts based on the low-level image segmentations. Thereby, information about lanes and boundaries can potentially be used for navigation planning. Through advancements in deep learning we have now come to a point where even reasoning of the space behind occluded parts of the images is possible for inferring road geometries [14]. In future work, we also plan to extent our work in this direction.

3) Deep Networks: All of above mentioned methods require handcrafted features/probabilities in some way to optimise the graph. It has been shown by now that deep networks with learned feature maps achieve much better semantic (instance) segmentation [1], [2] and thus understanding of the scene. Besides, they are able to generalise better when auxiliary output tasks are employed [15]. How-

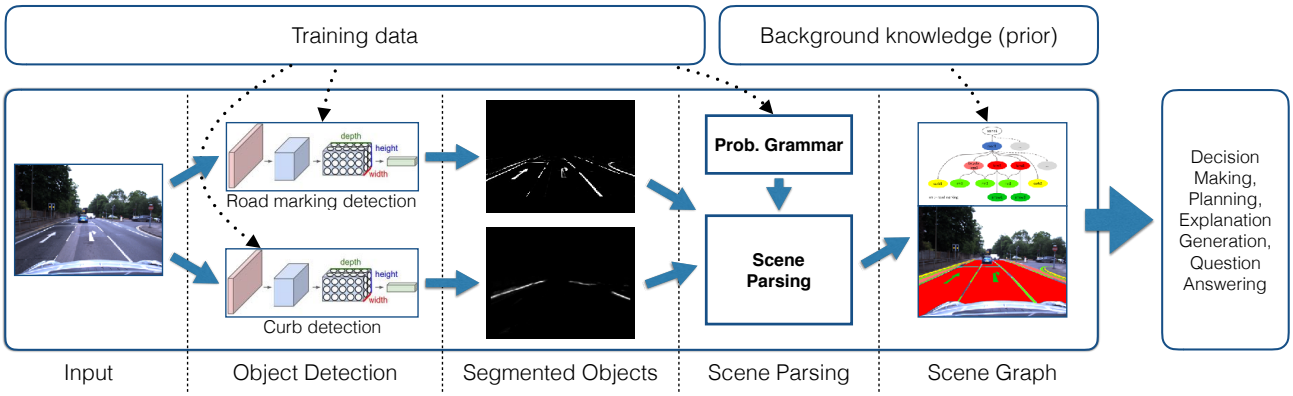


Fig. 2. Scene parsing approach based on road marking and curb detections. The approach has two main steps: (1) given an image, road marking and curb segments are detected by deep networks, and (2) given a set of detected segments, the scene is parsed using an adapted version of the Earley algorithm and a learnt probabilistic grammar. The resulting scene graph is integrated with domain knowledge and can used for planning and decision making.

ever, these networks suffer from interpretation and debugging difficulties and often fail to include prior information, high-level reasoning, or constraints (in the output space). Recently, some works have tried to improve some of these disadvantages by introducing spatial and semantic reasoning frameworks that can be trained in an end-to-end way [16]–[18]. In this work, we simply use deep networks as an effective way for segmenting an input image. However, our future goal is to extend this approach and to feed geometric, spatial, and semantic constraints back to the deep networks during learning.

III. Approach Overview

Our approach constructs a symbolic, graph-based description of the road layout given an image of a road scene (see Fig. 1). When interpreting the image, our approach considers two types of information: object detections and common road configurations based on learnt prior models.

Fig. 2 depicts the overall pipeline of our approach. We first segment the image by detecting curbs and road markings using trained deep networks (Sec. IV). These pixel-wise segmented images are clustered and the resulting entities are considered as input for a parsing process which generates a hierarchical scene representation (*scene graph*) (Sec. V). The parser takes object detections (and their uncertainty) and prior information of road scenes into account. Our probabilistic approach is in particular suitable for integrating incomplete and uncertain information from object detection pipelines. Each valid parse tree is scored by a probability which allows us to disambiguate between alternative representations. Intuitively, the score captures three aspects: (1) hierarchy (2) geometric features of detected entities, and (3) spatial relations between entities in the hierarchy. As we represent scene graphs using logical representations they can be linked to background knowledge and used for auditable planning and decision making.

IV. Scene Perception

This section describes how road markings and curbs are detected in a given image of a road scene. The resulting pixel-based images are clustered and segmented entities are obtained which are considered as input for the scene interpretation process described in Sec. V.

A. Road Marking Detection

Road markings are a critical component for (autonomous) driving especially in urban environments. The road rules are captured by their underlying meaning and they guide all traffic participants through potentially dangerous situations. Therefore, real-time detection and interpretation of road markings is an important cue for high-level scene understanding and aids planning and decision making.

Detecting all painted road markings (not just lane separators) on the road surface, which dictate the traffic rules for that particular urban setting, is a challenging problem for several reasons. Firstly, there are visual challenges such as occlusions, varying lighting, and changing weather conditions. Secondly, road markings vary from country to country and are often degraded. Lastly, there are no large datasets available for training with accurate ground-truth labels for road markings.

Road marking detection in images can be seen as a semantic segmentation problem. State-of-the-art methods for these tasks implement deep networks, which are able to learn specific scene context and thereby cope with the challenges stated above, as long as sufficient training data is available. Manually generating training data is extremely labour expensive, because of the required pixel-level detail in combination with the aforementioned visual issues. Therefore, we create road marking annotations in a weakly-supervised way, by leveraging complementary sensor modalities (i.e. LiDAR).

For generating the annotations, we exploit the property that road markings are highly reflective and must lie on the road surface. Firstly, we utilise the LiDAR point cloud to coarsely segment the road surface from the image. A dense CRF is then optimised to identify the road marking image pixels by

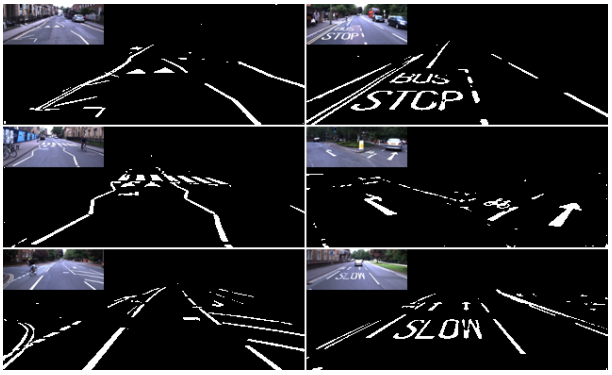


Fig. 3. Road marking detection performed by a deep semantic segmentation network in real-time. Before the detections are employed to generate the scene graph, they are mapped to top-down view.

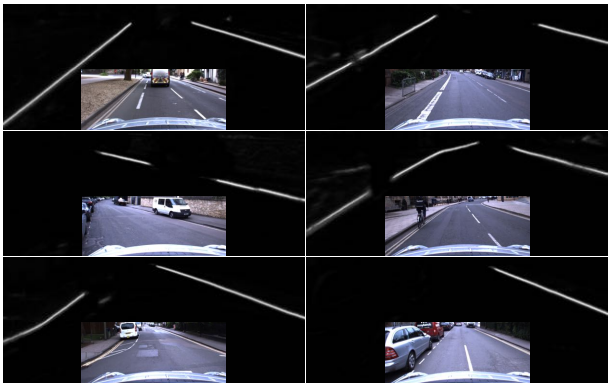


Fig. 4. Curbs are detected by a fully convolutional network. The network can detect visible curbs without making any assumptions about their 3D structure, shape or appearance.

corresponding them with the high-reflectance LiDAR points, which are not affected by varying lighting.

We employ these annotations to train a deep semantic segmentation network (inspired by U-net [19]) for road marking detection using only a monocular camera. The results demonstrate that the network segments the road markings from the image without any preprocessing steps, as shown in Fig. 3.

We direct the reader to [20] for a more detailed description of this method.

B. Curb Detection

Curbs (road boundaries) play an important role for autonomous cars as they intentionally and legally delimit driveable space. Curb detection using monocular images is a challenging problem. Road boundaries have narrow and long shapes which are not easily detectable. Deep networks often require large amounts of training data to obtain high-performance, well-generalised models. Due to colour, appearance, shape, perspective, illumination and background clutter, the training data should incorporate great variability changes. However, image by image hand labelling of the ground truth data is a time-consuming process. To avoid this problem and obtain a large amount of training samples, we generated 3D points cloud from 2D laser data and annotated points in the point cloud corresponding to road boundaries. Note that the 2D laser is attached vertically to the rear of a test car, which makes road boundaries easy to spot and annotate in the point cloud. The annotated points are projected to images of forward facing camera of the car. Lines are drawn between consecutive points to annotate road boundaries in-between the points. This way, hundreds of labelled images are obtained within an hour (approximately 750 images). A 10 kilometres dataset from the Oxford RobotCar Dataset introduced by [21] was annotated to generate several thousand semi-annotated masks. A vision based localiser was used to boost the number of training images by projecting labels from the annotated dataset to other traversals. However, some of the generated curbs masks contain annotations for occluded areas of curbs, such as over parked cars. To remove those redundant annotations, we trained U-net [19] with the raw masks and then run the inference with RGB images from the training data to generated output of detected curbs. The trained U-Net model can segment visible areas of curbs, but produces blurry outputs over occluding obstacles. Applying a threshold to the outputs gives us masks for detected visible curbs. We obtain labels for visible curbs by applying an AND operation between the thresholded outputs and raw labels. Finally, we train the U-net with visible curbs only (Fig. 4). A detailed description of our work on curb detection is given in [22].

V. Scene Interpretation

In the previous section, we explained how an input image is segmented into two classes: road markings and curbs. Before we describe how we learn a probabilistic grammar to parse these segmentations and construct a scene graph from them, we first introduce scene graphs formally.

A. Representation

Our motivation with this work is to support autonomous vehicles in their decision making, planning, and explanation generation. In particular, we aim at a representation that is interpretable (by machines and humans alike), extendable, and suitable for different inference tasks. To this end, we introduce *scene graphs* as a way to represent road scenes semantically using well-defined concepts and relations which are grounded in the vehicle's perception system.

Formally, scene graphs are represented in Description Logic; an overview is given in [23]. A scene is described by a set of instances of meaningful classes and their relations. For example, a *scene* is composed of a *road* which has two *curbs* and several *lanes* which in turn are bounded by several *road markings*. This hierarchical decomposition of a scene is important as we will explain later in Sec. V-C. In general, however, scene graphs can be linked flexibly to other information resources due to its underlying logical representation as we have shown in previous work [24]. For example, they can be linked to the outcome of detection and tracking algorithms of traffic participants and/or domain knowledge

TABLE I
SCENE GRAPH TAXONOMY

Class	Description
Scene	Root node of a scene graph. A *Scene* has at least one road (*Road*), but can have multiple.
Road	A road is delimited by at most two curbs (*Curb*) and has one or more lanes (*Lane*).
Curb	A curb is composed of one or multiple curb segments (*CurbSeg*).
Lane	A lane is bounded by road markings along the carriage way (*RMAlong*). Additionally, lanes can have road markings that are across the carriage way (*RMAcross*), and other road markings such as symbols and text (*RMOther*).
RMAlong	Road marking along the carriage way.
RMAcross	Road marking across the carriage way.
RMOther	Road marking of a symbol or text.
RMSeg	A road marking segment is a set of clustered pixels detected by the network described in Sec. IV-A. It can be one of three types: *RMALong*, *RMAcross*, or *RMOther*.
CurbSeg	A curb segment is a set of clustered pixels detected by the network described in Sec. IV-B.

TABLE II
LEARNT PROBABILISTIC CONTEXT-FREE GRAMMAR

Production rule
Scene $\rightarrow$ *Road*
Road $\rightarrow$ *Curb Lane*
Lane $\rightarrow$ *RMAlong RMAcross RMOther*
RMAlongCW $\rightarrow$ *RMSeg*
RMAcrossCW $\rightarrow$ *RMSeg*
RMOther $\rightarrow$ *RMSeg*
Curb $\rightarrow$ *CurbSeg*

defined by the Highway Code. This kind of knowledge can be encoded as logical rules within Description Logic.

A brief description of the most important concepts is given in Tab. I. It is important to note that entities that represent road marking segments (*RMSeg*) and curb segments (*CurbSeg*) are both linked to the output of the segmentation networks described in the previous section. Hence, instances of these types are grounded in image space. This is important as it allows us to reconstruct concepts higher-up in the hierarchy (e.g. Lanes) based on those low-level segmentations. In particular, we represent detected segments using axis-aligned and minimal area bounding boxes. More high-level concepts are represented as the bounding box of their children. Note that all other concepts are assigned based on the learnt grammar.

In the next section, we explain how we learn a probabilistic grammar for road scenes based on the introduced concepts.

B. Probabilistic Grammar

We adopt the approach by [11] and learn a probabilistic context-free grammar for road scenes from a set of annotated examples. To this end, we consider a set of scene graphs that have been manually annotated according to the concepts introduced in the previous section and based on the detections of road markings and curbs (Sec. IV). We learn the structure of the production rules and their probability from the frequency observed in the annotated set. The production rules are shown in Tab. II[1]

For each annotated scene graph we compute a set of geometric properties and spatial relations between instances that share the same parent node. We start the computation at the leaf nodes and propagate the results up the hierarchy. In our implementation, we consider several geometric properties including: length, width, and area for both axis-aligned and minimal bounding boxes. Furthermore, we consider the ratios between these properties to compute scores for the *axis-alignedness*, *alongness*, and *acrossness* of an instance. We also consider spatial relations between instances that share the same parent node (e.g. two boundaries of a lane). For these instances, we compute several relations including: the connectivity of the bounding boxes based on the Region Connection Calculus [25] and their relative angle and distance based on the Ternary Point Calculus [26]. In total we consider 18 geometric properties and 14 spatial relations. However, the details of how these properties and relations are not described here for brevity. Overall, the individual features are not critically important (and can be replaced). However, they provide us with the ability to assess the overall probability of the scene by considering all instances of a tree t given its geometric description and its relations. For each geometric property and relation we learn a probability distribution, namely $P_{geo}(x)$ and $P_{rel}(x)$, based on the annotated data using Kernel Density Estimation (based on Gaussian kernels). By computing the probability of each individual property and relation we can compute the overall probability of a tree based on the grounded representation as follows:

$$P(s|t,g) = \prod_{x \in t} P_{geo}(x) P_{rel}(x) \tag{1}$$

whereby s denotes a scene, t a tree, and g a grammar.

C. Scene Parsing

To reconstruct the layout of a road scene we use an extended version of a probabilistic Earley parser [27]. In general, the Earley algorithm is a dynamic programming approach that is able to handle ambiguous grammars. It combines top-down predictions and bottom-up recognitions to effectively parse its input. The algorithm has three main steps: *predict*, *scan*, and *complete*. In the predict step, rules are expanded according to the grammar. This step guides the overall search in a top-down way (initially the root node is expanded). In the scan step, the next input symbol is read and compared to the next one that was predicted. If a production rule is completed, the complete step has found a valid parse of a subtree and overall search is advanced. This type of

[1]Note, that we have omitted the learnt probabilities as we have learn different rules for different cardinalities.

hybrid search using top-down down reasoning and bottom-up perception for scene understanding can be very effective in real-world scenarios as we have shown earlier [28].

Our adapted version of the parser takes the learnt probabilistic grammar and a sequence of curb and road marking segments as input. The segments form the lexicon of our grammar and their probabilities are determined according to $P_{geo}(X)$ as defined in the previous section.

After the parser has recognised the input, a forest of parse trees can be retrieved. In our implementation we use a shared packed parse forest (SPPF) to store the ambiguous parse trees [29]. Parse trees are evaluated according their probabilities computed as follows:

$$P(t|s,g) = P(t|g)P(s|t,g) \tag{2}$$

whereby t denotes a parse tree, s the scene, and g the grammar. $P(t|g)$ is the product of all probabilities according to the production rules and $P(s|t,g)$ represents the data likelihood of seeing this scene given the tree and the grammar. Eventually, the best parse tree t^* can be chosen according to the overall probability:.

$$t^* = \arg\max_{t \in \mathcal{T}} P(t|s,g) \tag{3}$$

whereby t denotes a parse tree in the parse forest $\mathcal{T}$, s the scene, and g the grammar.

VI. Experiments

In this section we present the experimental setup and discuss qualitative results of our approach.

A. Experimental Setup

In this work, we evaluated the overall pipeline as depicted in Fig. 2. A given input image is processed by the road marking and the curb detection networks. The output of these networks is a probability distribution of segments in the image space. Using Inverse Perspective Mapping (IPM), we transform each of the segmented images into a birds eye view (see Tab. III). For each class, we then find clusters that represent these entities by their bounding boxes and compute a set of geometric features. Based on their visual and geometric probability these segmented entities are added to the lexicon of the grammar.

The Earley algorithm predicts the structure of the scene based on the learnt grammar and parses the segments from left to right in image space. We evaluated the generated parse trees according to their probability. However, given the high ambiguity of rules in the learnt grammar, we have selected a few examples manually (Tab. III). In the next section we discuss several theses examples and point to interesting and/or problematic aspects.

B. Qualitative Results

Tab. III depicts the qualitative results for several scenes. The table shows the input image; the different segments produced by the networks and the clustering step (road markings in green; curbs in orange); and the generated scene graphs (or parts of it).

Scene (a) In this scene (see Fig. 1), the segmentation captures curbs on both sides of the road as well as road markings along the carriage way. However, a stop line as well as the bicycle symbol are not detected. By integrating some domain knowledge from the Highway Code in form of rules, we can refine the scene graph by inferring that there is a bicycle lane on the left-hand side as the lane's width is too narrow for a standard car lane. These rules are encoded within Description Logic and can infer classes which were not labelled in any of the examples. However, we are not able to infer the same on the right-hand side as we do not have any meaningful segment that describes the boundary of the bicycle lane on the right-hand side. The detection of road markings and curbs in roads other than the main road is typically more challenging as they are perceived at the edge of the camera's field of view.

Scene (b) In this scene the parser detects two road markings on a lane. Given their size and spatial relation we can infer that these entities are road markings that introduce speed humps on the road.

Scene (c) This scene is interesting as there are curb structures in the middle of the road. Furthermore, the left lane has two stop-lines. However, it is important for an autonomous vehicle to infer that is has to stop in front of the first one. Note, that such an inference can only be drawn when local context of the scene is considered, but not from the single segment alone. These are situations in which we believe that background knowledge and AI reasoning techniques can have a great impact when interpreting scenes.

Scene (d) In this scene both curbs and road markings are well detected (except for the degraded dotted line across the road). However, this scene provides an interesting and rare case as the road markings for the car (zig-zag line) and the bicycle lane overlap. Momentarily this cannot be represented by our grammar as we made the assumption that lanes are next to each other.

In future work we will also perform a quantitative analysis of our approach, in particular with respects to its real-time capabilities. In general the Early algorithm is well-suited for real-time applications as its worst time complexity is $O(n^3)$. However, retrieving and processing a potential exponential number of parse trees might be challenging.

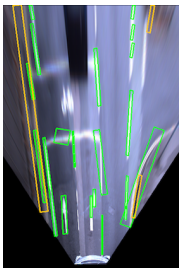
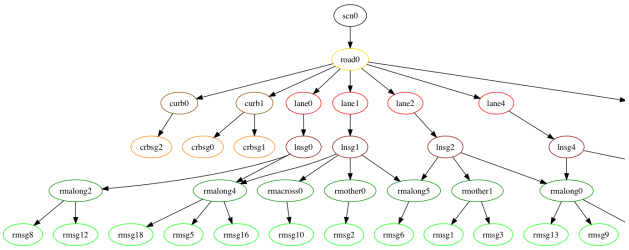
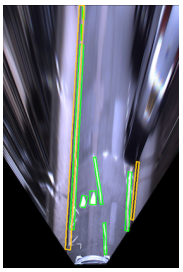
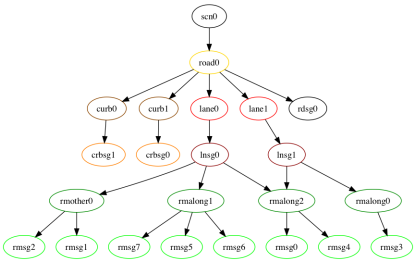
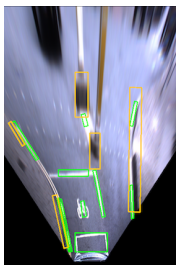
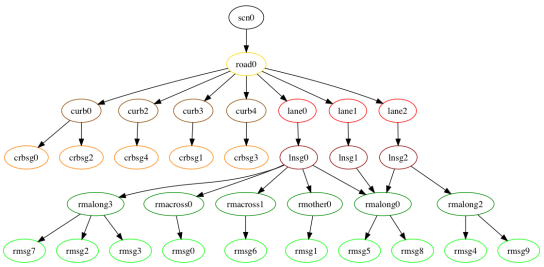
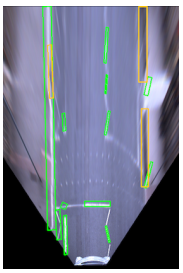
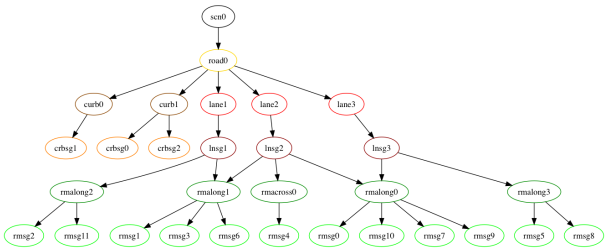
VII. Discussion

In this section we would like to provide a brief overview of the application space of the scene graph.

1) Urban traffic scenes are highly structured since they are built consistently according to specified road rules. By incorporating these rules, certain nodes in the scene graph can be classified. For instance, a bicycle lane is easily distinguished from a car lane by comparing the width. In this way, the scene graph allows for classification of road objects/segments without requiring expensive manual labels.

2) The segmented scenes given by the scene graph can be employed to bootstrap deep learning models. As stated above classification labels which can be used for training

TABLE III
QUALITATIVE RESULTS

ID	Original (RGB)	Segments (IPM)	Scene Graph (partial)
(a)			
(b)			
(c)			
(d)			

purposes can be acquired without expensive manual annotation. Furthermore, the scene graph provides an informed indication about the likely location of road objects (e.g. curbs, road markings). This could be used when training deep networks for instance to guide attention or to adjust the loss and thereby improve performance. In this way, important prior information about the environment is included in a deep learning approach (which is non-trivial).

3) Scene graphs can be used for (cost-based) planning for autonomous vehicles as they reason about the lane geometry and can infer road marking classes based on contextual spatial relations. For instance, a solid boundary of a bicycle lane should only be crossed in case of emergency. Besides, actions are now interpretable because we can review the representation inferred from the segmentation.

4) The scene graph is able to predict/hallucinate missing objects because of the learned spatial and semantic constraints. For example, two-way roads with missing lane markings in the middle will not fit the learned representations (nor the road rules). The scene graph can predict the most likely lane geometry in that case.

We think that these examples are interesting uses cases with exciting technological challenges for applications of scene graphs.

VIII. CONCLUSION

In this paper we presented an approach for scene understanding of complex urban environments. To this end, we

proposed *scene graph*, a hierarchical, graph-based representation, and a parsing pipeline that generates and evaluates scenes graphs based on partially segmented images, a learnt probabilistic grammar, as well as geometric and relational models. Furthermore, we have presented and discussed several example scenarios in which scene graphs can provide meaningful insights in the overall structure of the environment. The construction and interpretation of interpretable and auditable scene graphs can play essential role in many tasks of autonomous vehicles including planning, decision making, and explanation generation. Hence we believe that this functionality can have wide impact in the context of autonomous driving and mobile robotics in general.

ACKNOWLEDGMENT

The work has been supported by the EPSRC/UK Research and Innovation Programme Grant EP/M019918/1 (Mobile Autonomy: Enabling a Pervasive Technology of the Future).

We acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

REFERENCES

[1] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *arXiv preprint arXiv:1511.00561*, 2015.

[2] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2881–2890.

[3] A. Geiger, M. Lauer, C. Wojek, C. Stiller, and R. Urtasun, "3d traffic scene understanding from movable platforms," *IEEE transactions on pattern analysis and machine intelligence*, vol. 36, no. 5, pp. 1012–1025, 2014.

[4] J. Ziegler, P. Bender, M. Schreiber, H. Lategahn, T. Strauss, C. Stiller, T. Dang, U. Franke, N. Appenrodt, C. G. Keller *et al.*, "Making bertha drive an autonomous journey on a historic route," *IEEE Intelligent Transportation Systems Magazine*, vol. 6, no. 2, pp. 8–20, 2014.

[5] J.-B. Bordes, F. Davoine, P. Xu, and T. Denœux, "Evidential grammars: A compositional approach for scene understanding. application to multimodal street data," *Applied Soft Computing*, vol. 61, pp. 1173–1185, 2017.

[6] F. Dierkes, M. Raaijmakers, M. T. Schmidt, M. E. Bouzouraa, U. Hofmann, and M. Maurer, "Towards a multi-hypothesis road representation for automated driving," in *Intelligent Transportation Systems (ITSC), 2015 IEEE 18th International Conference on*. IEEE, 2015, pp. 2497–2504.

[7] D. Töpfer, J. Spehr, J. Effertz, and C. Stiller, "Efficient road scene understanding for intelligent vehicles using compositional hierarchical models," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 1, pp. 441–451, 2015.

[8] J. Spehr, D. Rosebrock, D. Mossau, R. Auer, S. Brosig, and F. M. Wahl, "Hierarchical scene understanding for intelligent vehicles," in *Intelligent Vehicles Symposium (IV), 2011 IEEE*. IEEE, 2011, pp. 1142–1147.

[9] S. Kashetty Venkateshkumar, M. Sridhar, and P. Ott, "Latent hierarchical part based models for road scene understanding," in *The IEEE International Conference on Computer Vision (ICCV) Workshops*, December 2015.

[10] B. Mathibela, P. Newman, and I. Posner, "Reading the road: Road marking classification and interpretation," *IEEE Trans. Intelligent Transportation Systems*, vol. 16, no. 4, pp. 2072–2081, 2015. [Online]. Available: http://dx.doi.org/10.1109/TITS.2015.2393715

[11] T. Liu, S. Chaudhuri, V. G. Kim, Q.-X. Huang, N. J. Mitra, and T. Funkhouser, "Creating consistent scene graphs using a probabilistic grammar," *ACM Transactions on Graphics (Proc. SIGGRAPH Asia)*, vol. 33, no. 6, Dec. 2014.

[12] A. Ess, T. Mueller, H. Grabner, and L. van Gool, "Segmentation-based urban traffic scene understanding," in *Proceedings of the British Machine Conference, pages*, 2009, pp. 84–1.

[13] A. Joshi and M. R. James, "Generation of accurate lane-level maps from coarse prior maps and lidar," *IEEE Intelligent Transportation Systems Magazine*, vol. 7, no. 1, pp. 19–29, 2015.

[14] S. Schulter, M. Zhai, N. Jacobs, and M. Chandraker, "Learning to look around objects for top-view representations of outdoor scenes," *arXiv preprint arXiv:1803.10870*, 2018.

[15] A. Kendall, Y. Gal, and R. Cipolla, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," *arXiv preprint arXiv:1705.07115*, 2017.

[16] X. Chen, L.-J. Li, L. Fei-Fei, and A. Gupta, "Iterative visual reasoning beyond convolutions," *arXiv preprint arXiv:1803.11189*, 2018.

[17] X. Liang, H. Zhou, and E. Xing, "Dynamic-structured semantic propagation network," *arXiv preprint arXiv:1803.06067*, 2018.

[18] N. Nauata, H. Hu, G.-T. Zhou, Z. Deng, Z. Liao, and G. Mori, "Structured label inference for visual understanding," *arXiv preprint arXiv:1802.06459*, 2018.

[19] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2015, pp. 234–241.

[20] T. Bruls, W. Maddern, A. A. Morye, and P. Newman, "Mark yourself: Road marking segmentation via weakly-supervised annotations from multimodal data," in *Robotics and Automation (ICRA), 2018 IEEE International Conference on*. IEEE, 2018.

[21] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 Year, 1000km: The Oxford RobotCar Dataset," *The International Journal of Robotics Research (IJRR)*, vol. 36, no. 1, pp. 3–15, 2017. [Online]. Available: http://dx.doi.org/10.1177/0278364916679498

[22] T. Suleymanov, P. Amayo, and P. Newman, "Inferring road boundaries through and despite traffic," in *The 21st IEEE International Conference on Intelligent Transportation Systems*, November 2018.

[23] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, Eds., *The Description Logic Handbook: Theory, Implementation, and Applications*. New York, NY, USA: Cambridge University Press, 2003.

[24] M. Tenorth, L. Kunze, D. Jain, and M. Beetz, "Knowrob-map - knowledge-linked semantic object maps," in *2010 10th IEEE-RAS International Conference on Humanoid Robots*, Dec 2010, pp. 430–435.

[25] D. A. Randell, Z. Cui, and A. G. Cohn, "A spatial logic based on regions and connection," in *KR*. Morgan Kaufmann, 1992, pp. 165–176.

[26] R. Moratz and M. Ragni, "Qualitative spatial reasoning about relative point position," *Journal of Visual Languages & Computing*, vol. 19, no. 1, pp. 75–98, 2008, spatial and Image-based Information Systems. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1045926X06000723

[27] J. Earley, "An efficient context-free parsing algorithm," *Commun. ACM*, vol. 13, no. 2, pp. 94–102, Feb. 1970. [Online]. Available: http://doi.acm.org/10.1145/362007.362035

[28] L. Kunze, C. Burbridge, M. Alberti, A. Tippur, J. Folkesson, P. Jensfelt, and N. Hawes, "Combining top-down spatial reasoning and bottom-up object class recognition for scene understanding," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Chicago, Illinois, US, September, 14–18 2014.

[29] E. Scott, "Sppf-style parsing from earley recognisers," *Electronic Notes in Theoretical Computer Science*, vol. 203, no. 2, pp. 53 – 67, 2008, proceedings of the Seventh Workshop on Language Descriptions, Tools, and Applications (LDTA 2007). [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1571066108001497

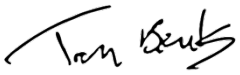
Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor **(only required where there isn't already a statement of contribution within the paper itself).**

Title of Paper	Reading between the Lanes: Road Layout Reconstruction from Partially Segmented Scenes
Publication Status	Published
Publication Details	L. Kunze, **T. Bruls**, T. Suleymanov, and P. Newman, "Reading between the lanes: Road layout reconstruction from partially segmented scenes", in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Nov. 2018, pp. 401-408.

Student Confirmation

Student Name:	Tom Adriaan Hubert Bruls		
Contribution to the Paper	Contributions included: - Refining the initial ideas regarding the hybrid framework. - Preparing the dataset for road marking segmentation. - Running the road marking segmentation experiments. - Writing the related work and road marking detection section. - Creating the road marking detection figure. The overall paper emerged as a product of discussions and collaboration with my co-authors.		
Signature		Date	10-05-2020

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Professor Paul Newman			
Supervisor comments			
Signature		Date	11-05-2020

This completed form should be included in the thesis, at the end of the relevant chapter.

B

I Can See Clearly Now: Image Restoration via De-Raining

This appendix contains a reproduction of the following publication:

> [22] H. Porav, **T. Bruls**, and P. Newman, "I can see clearly now: Image restoration via de-raining", in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 7087-7093[1].

[1] video accompanying the paper with extensive explanations and results: `https://www.youtube.com/watch?v=P4a7C-V70Y8`

I Can See Clearly Now : Image Restoration via De-Raining

Horia Porav, Tom Bruls and Paul Newman

***Abstract*— We present a method for improving segmentation tasks on images affected by adherent rain drops and streaks. We introduce a novel stereo dataset recorded using a system that allows one lens to be affected by real water droplets while keeping the other lens clear. We train a denoising generator using this dataset and show that it is effective at removing the effect of real water droplets, in the context of image reconstruction and road marking segmentation. To further test our de-noising approach, we describe a method of adding computer-generated adherent water droplets and streaks to any images, and use this technique as a proxy to demonstrate the effectiveness of our model in the context of general semantic segmentation. We benchmark our results using the CamVid road marking segmentation dataset, Cityscapes semantic segmentation datasets and our own real-rain dataset, and show significant improvement on all tasks.**

I. Introduction

If we want machines to work outdoors and see while doing so, they have to work in the rain. When rain and lenses interact, computer vision becomes harder - wild local distortions of the image appear which dramatically impede image understanding tasks. However the distortions are not noise, they are structured, the light field is simply bent and attenuated, and accordingly can be modelled and reversed.

In this work we develop a filter which as a pre-processing step removes the effect of raindrops on lenses. Several tasks are affected by the presence of adherent water droplets on camera lenses or enclosures, such as semantic segmentation [1], localisation using segmentation [2], [3] or road marking segmentation [4]. In this paper we choose to use segmentation as an example task by which to test the effectiveness of our method. Many approaches so far have reached for multi-modal data [5], domain adaptation [6], [7] or training on synthetic data [8], however this can become awkward as:

1) Acquiring rainy images is time-consuming, expensive or impossible for many tasks or setups, especially in the case of supervised training, where ground truth data is needed.
2) Training, domain-adapting or fine-tuning each individual task with augmented data is intractable.

We take a different approach and build a system as an image preprocessor, the output of which is a cleaned, de-rained image that improves the performance of many tasks performed on the image.

We begin by creating a bespoke real-world small baseline stereo dataset where one lens is affected by real water droplets and the other is kept dry. The methodology and apparatus for doing so is presented in section IV-A. Using this dataset, we train a de-raining generator and show that it is able to both drastically improve the visual quality of images and restore performance on road marking segmentation tasks.

Authors are from the Oxford Robotics Institute, University of Oxford, UK. {horia,tombruls,pnewman}@robots.ox.ac.uk

Fig. 1. We learn a de-noising generator that can remove noise and artefacts induced by the presence of adherent rain droplets and streaks. On the top left,input images that are affected by real rain drops. On the top right, the cleaned, de-rained images. On the bottom left, input images that are affected by computer-generated rain drops. On the bottom right, the cleaned, de-rained images.

Secondly, we describe a way of efficiently adding computer-generated adherent rain droplets and adherent streaks to any image using GPU shaders. This system is presented in section III-A. As the Cityscapes dataset provides a good groundtruth for segmentation but does not contain images with significant rain on the lens, we modify it using this technique and use it as a proxy to study the effects of rain on general semantic segmentation. Additionally, we create a synthetic rain dataset by adding computer-generated rain drops to a full Oxford RobotCar dataset [9] and to the CamVid [10] dataset.

Our main contributions include:

- a de-raining model that produces state of the art results;
- using computer-generated water drops as a proxy to study the effects of rain on segmentation for datasets that provide a ground truth but do not normally contain rainy images; and
- a real-world very-narrow-baseline stereo dataset with rainy & clear images covering a wide array of dynamic scenes.

Our aim is to show that pre-processing the image leads to better performance as compared to training, retraining or fine-tuning a task-specific model with rain-augmented data.

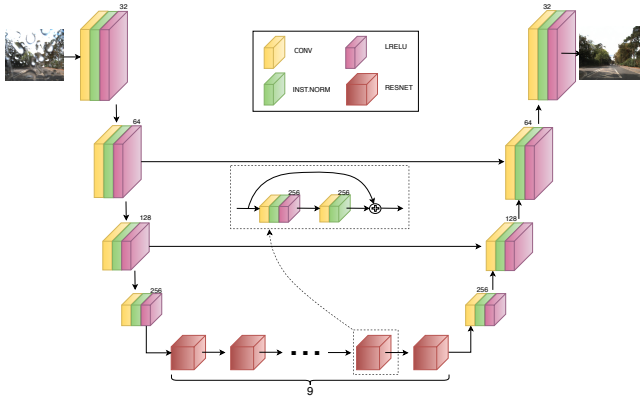


Fig. 2. The internal architecture of our generator. We motivate the addition of additive skip connections by observing that much of the structure of the input image should be kept, along with illumination levels and fine details.

We benchmark our de-raining model on the following tasks:

- Road marking segmentation and image restoration on a real-world small baseline stereo dataset where one lens is affected by real water droplets and the other is kept dry and clear.
- Image reconstruction on the real-world dataset of [11].
- Road marking segmentation and image restoration on CamVid [10] and RobotCar [9] imagery with computer-generated droplets added.
- Semantic segmentation on Cityscapes [1] imagery with computer-generated droplets added.

The quantitative and qualitative results are presented in section V.

II. Related Work

Generally speaking, the quality of an image can be affected in two ways by bad weather conditions. Firstly, contaminants in the atmosphere, such as falling rain, fog, smog or snow will hinder visibility or partially occlude a scene but do not significantly distort the image. Secondly, adherent contaminants such as water droplets, which stick to transparent surfaces or lenses, tend to heavily distort the image, essentially acting as a secondary lens with various degrees of blurring. Several techniques are employed to clean the first type of images, such as those used by [12], [13], [14], [15], [16], however these techniques cannot be used to restore images affected by adherent rain, as the optics involved differ significantly from those of atmospheric droplets. The remainder of this section outlines some of the techniques used to tackle the effects of adherent rain droplets and adherent streaks.

Rain Modelling and Simulation: In the context of computer vision, several studies have attempted to model the structure and optical properties of adherent water droplets. The authors of RIGSEC [17], [18] model raindrops first as sections of a sphere and later account for the effect of gravity using 2D Bezier curves, and confirm experimentally that a physically correct droplet shape can be computed using this method. [19] additionally study and model the dark band around the edges of adherent drops, and show that a simplified model is enough to correctly undistort the image on the surface of the droplet.

We base our simple synthetic droplet model on the works of [17], [18] and [19], by storing proto-droplet normal maps which are subsequently warped and combined at run time using an approach similar to meta-balls [20].

Additionally, several small datasets have been created to benchmark the accuracy of de-raining techniques. In [21], water is sprayed on a glass pane fitted in front of a camera, but no ground truth is provided due to temporal illumination and scene changes. A video sequence where the lens is affected by real rain droplets is also provided, again without ground truth. The authors of [22] again use a glass pane sprayed with water to study the performance of their droplet detection and removal pipeline, but only offer ground truth for the position of the droplets. The first attempt to provide accurate ground truth is made by [11], in which images of static scenes are captured both with and without a glass pane sprayed with water in front of the camera. This process is, however, very difficult to scale to the number of images required by modern deep-learning approaches. To our best knowledge, we are the first to record a real-world large dataset of sequential dynamic scenes with an accurate, clear ground truth and a large variation in raindrop type and size.

Raindrop Detection and Removal: In [17] and [18], raindrops are detected by attempting to match a template of a synthetic raindrop at locations where the presence of a real drop is hypothesized. This approach breaks down when the shape of the real droplets differs significantly from that of the template. The authors of [22] take a different approach by observing that the motion inside droplets is between 1/30 and 1/20 slower than that in the scene. They use this information to detect raindrops and then attempt to restore the image by using a combination between image inpainting and recovering data from within the distorted image formed on the droplet. Both techniques use multi-frame information for image reconstruction, and are not applicable to single-images.

Multi-camera and pan-tilt setups are exploited by [23], [24], [25] and [26]. These techniques use disparities to detect droplets and subsequently attempt to replace the affected regions in one lens with information from the other lens. This approach does not work on single images and assumes that the same regions are not covered by rain in both frames.

Convolutional neural networks were used by [21] to restore images affected by dirt and rain. They use a simple 3-layer architecture, each with 512 units, which works well on small drops but breaks down with much larger contaminants. A much larger Generative Adversarial Network (GAN) model [27] is used by [11], along with attention [28]. They leverage their static dataset to provide a ground truth for the droplet attention mask and train a recurrent model that outputs a heatmap of the location of the droplets. This heatmap is then concatenated with the input image and run through the GAN. They produce state-of-the-art results and made their dataset publicly available, which has allowed us to directly compare our method with theirs.

III. Learning to Clean Images

A. Computer-Generated Synthetic Rain

We base our simple synthetic droplet model on the works of [17], [18] and [19], generate the locations of raindrops using a simple statistical approach, model the interactions between raindrops using metaballs [20] and implement its rendering efficiently using GPU shaders.

A proto-raindrop is created using a simple refractive model that assumes a pinhole camera. The refraction angle is encoded following a scheme similar to normal mapping [29] by using a 2D look-up table represented by the RED and GREEN channels of a texture T, with the thickness of the drop encoded in the BLUE channel of the same texture. This texture T is then masked using an alpha layer that allows blending of the water drops with the background image and other drops, as shown in Figure 3a. With the drop acting as a simple lens, the coordinate (x_r, y_r) of the world point that is rendered at the location (u, v) on the surface of a drop is given by the following simplified distortion model:

$$x_r = u + (R * B) \tag{1}$$
$$y_r = v + (G * B). \tag{2}$$

Each image location (u, v) has a probability P_r of becoming the center of a proto-raindrop whose dimensions are scaled along the horizontal and vertical directions by a tuple of random values S_x and S_y. For each timestep, the center of a droplet may undergo a slip of D_x pixels along the horizontal and D_y pixels along the vertical direction as a function of the droplet diameter d:

$$D_x, D_y = \begin{cases} 0, 0 & d \leq 4mm \\ x \sim \mathcal{N}(0, 3), P_d * 5 & d > 4mm, \end{cases}$$

where P_d represents the probability of slip along the vertical direction and x denotes the random deviation of the slip along the horizontal direction. We empirically choose a maximum of 5 pixels of vertical displacement.

For each timestep, droplets that are close to each other are merged using the metaballs approach [20], as shown in Figure 3b. By default, each texture location $T(u, v)$ that does not fall under a droplet encodes a normal that is perpendicular to the background image. Finally, the image is sampled using the normal map defined by the texture T to produce a result similar to the one in the top-left corner of Fig 1. Using this technique we have created three synthetic rain datasets:

- synthetic rain added to CamVid, complete with road marking ground truth;
- synthetic rain added to Cityscapes, complete with semantic segmentation ground truth; and
- synthetic rain added to the dry images from our stereo dataset, complete with road marking ground truth.

Fig. 3. Metaballs.

B. The de-raining network

The de-raining network architecture is based on Pix2PixHD [30]. The architecture is shown in Fig. 2. We employ 4 down-convolutional layers with stride 2, followed by 9 ResNet [31] blocks and 4 up-convolutional layers. We motivate the addition of skip connections by observing that most of the structure of the input image should be kept, along with illumination levels and fine details.

To promote better generalization and inpainting, we refrain from using any direct pixel-wise loss and instead use a combination of adversarial, perceptual, and multi-scale discriminator feature losses. The discriminator architecture is a CNN with 5 layers, similar to PatchGAN [32]. We present the full structure of the losses in the next section.

C. Losses

Similar to [33], we apply an adversarial loss through a discriminator on the output of the generator. This loss is formulated as:

$$\mathcal{L}_{adv} = (D(G(I_{\text{rainy}})) - 1)^2. \tag{3}$$

The discriminator is trained to minimize the following loss:

$$\mathcal{L}_{disc} = (D(I_{\text{clear}}) - 1)^2 + (D(I_{\text{de-rained}}))^2, \tag{4}$$

where I_{derained} is sampled from a pool of previously de-rained images.

The perceptual loss [34] is applied between the label and reconstructed image:

$$\mathcal{L}_{\text{perc}} = \sum_{i=1}^{n_{VGG}} \frac{1}{w_i^{perc}} \| VGG(I_{\text{clear}})_i - VGG(G(I_{\text{rainy}}))_i \|_1, \tag{5}$$

where n_{VGG} represents the number of VGG layers that are used to compute the loss and $w_i^{perc} = 2^{(n_{VGG}-i)}$ weighs the importance of each layer.

Additionally, a multi-scale discriminator feature loss [30] is applied between the label and reconstructed image:

$$\mathcal{L}_{\text{msadv}} = \sum_{i=1}^{n_{ADV}} \frac{1}{w_i^{adv}} \| D(I_{\text{clear}})_i - D(G(I_{\text{rainy}}))_i \|_1, \tag{6}$$

where n_{ADV} represents the number of discriminator layers that are used to compute the loss and $w_i^{adv} = 2^{(n_{ADV}-i)}$ weighs the importance of each layer.

The complete generator objective $\mathcal{L}_{\text{gen}}$ becomes:

$$\mathcal{L}_{\text{gen}} = \lambda_{\text{adv}} * \mathcal{L}_{\text{adv}} + \lambda_{\text{perc}} * \mathcal{L}_{\text{perc}} + \lambda_{\text{msadv}} * \mathcal{L}_{\text{msadv}}. \tag{7}$$

Each λ term is a hyperparameter that weights the importance of each term of the loss equation. We wish to estimate the generator G and discriminator D functions such that:

$$G, D = \arg\min_{G,D} \mathcal{L}_{\text{gen}} + \mathcal{L}_{\text{disc}}. \tag{8}$$

In the following section we describe how the network is trained to minimise the above losses.

IV. Experimental Setup

A. Stereo rain dataset

In this section we present the hardware used to record our narrow-baseline stereo dataset that allows one lens to be affected by real water droplets while keeping the other lens clear. The camera setup is shown in Figure 8. A 3D-printed bi-partite chamber is sandwiched between two acrylic clear panels and placed in front of the two lenses, with the left-hand section of the chamber being kept dry at all times, while

Fig. 4. CamVid road marking segmentation results. From left to right: rainy input image, segmentation result on rainy image, derained input image, segmentation result on derained image.

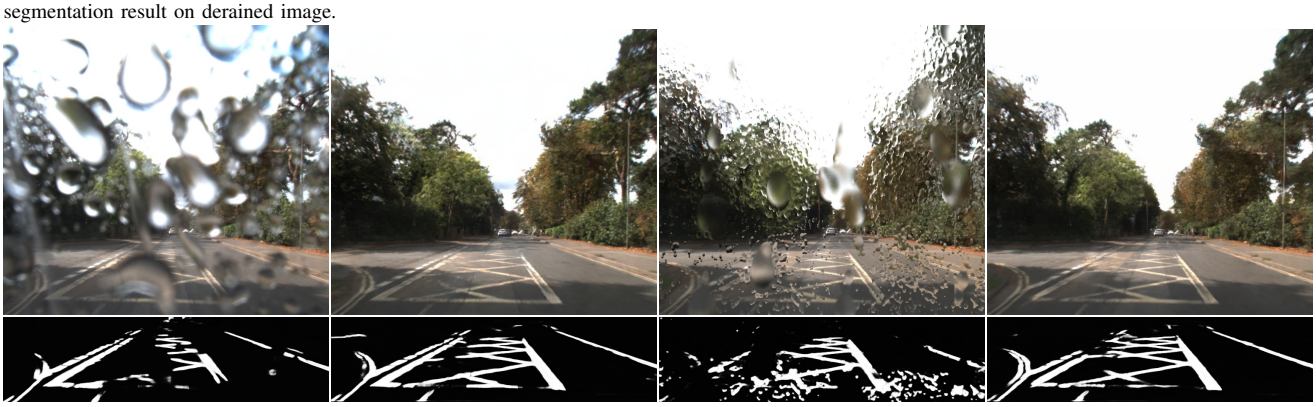


Fig. 5. RobotCar road marking segmentation results.First column shows a RobotCar(R) real rain image and segmentation result. Second column shows the derained real rain image and segmentation result. Third column shows a RobotCar(S) computer-generated rain image and segmentation result. Fourth column shows the derained computer-generated rain image and segmentation result.

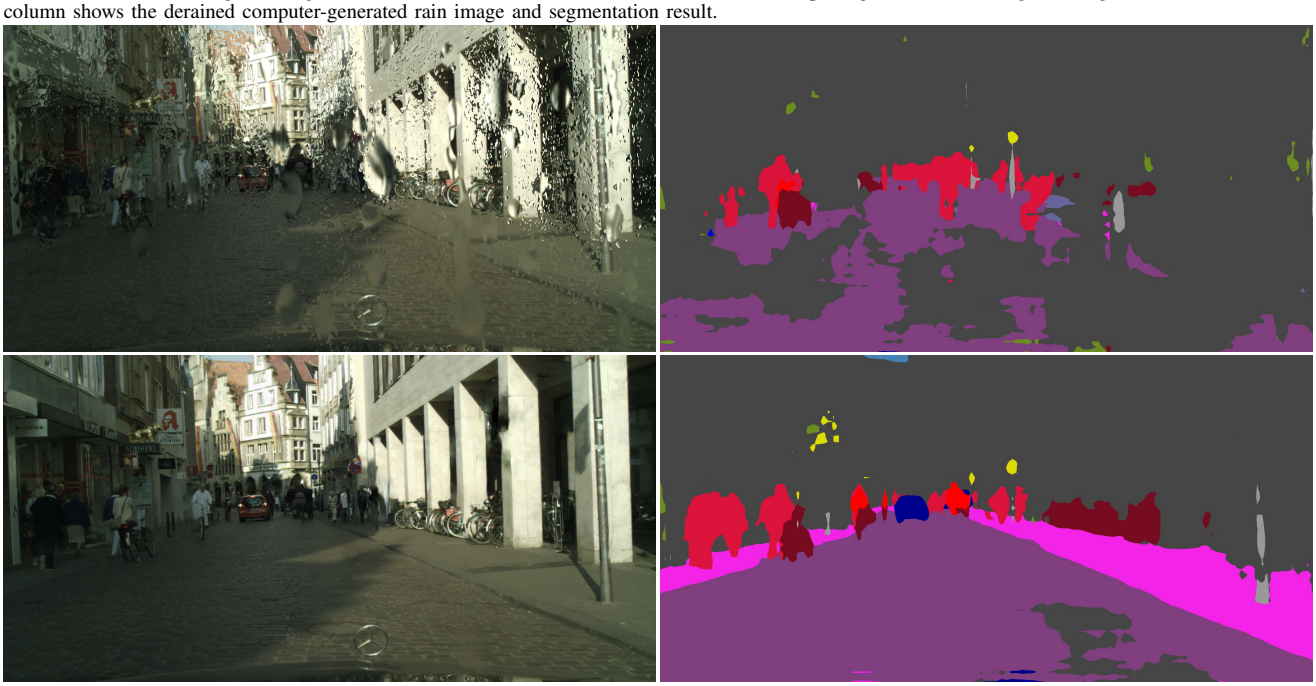


Fig. 6. Cityscapes semantic segmentation results. The first row shows a rainy image on the left and its corresponding semantic segmentation on the right. The second row shows the derained image on the left and its corresponding semantic segmentation on the right.

Fig. 7. An example from our stereo dataset. The image on the left is produced by the left lens, which is affected by water drops. The image in the middle is produced by the dry right hand lens. The image on the right is the road marking segmentation ground truth.

the right-hand section is sprayed with water droplets using an internal nozzle fitted at the top of the chamber. The angle of this chamber with respect to the axes of the cameras can be modified to simulate a slanted windscreen or enclosure, and the distance from the lenses can be increased or decreased accordingly to replicate different levels of focus or blur on the droplets.

The nozzle spans the entire width of the right chamber and is capable of producing water droplets with a diameter between 1mm and 8mm, as well as streaks of water. This variability is achieved by modulating the water pressure using a number of pulse width modulation regimes. The water is drained from the bottom of the chamber and is returned to a storage tank for recirculation. The cameras used are Point Grey Grasshopper 2 with 4.5 mm F/1.4 lenses, a baseline of 29 mm and automatic synchronisation. The system is fully portable and the water is completely contained within the circuit formed by the right chamber, pump and tank.

We have collected approximately 50000 pairs of images by driving in and around the city of Oxford. The image pairs are undistorted, cropped and aligned. We have selected 4818 image pairs to form a training, validation and testing dataset. From the testing partition, we have created ground truth road marking segmentations for 500 images. An example from our dataset is shown in Figure 7.

Compared to the painstakingly-collected dataset of [11], our setup is a set-and-forget approach: once the stereo camera has been mounted on a vehicle, it is trivial to collect large amounts of well-synchronised and well-aligned pairs of images.

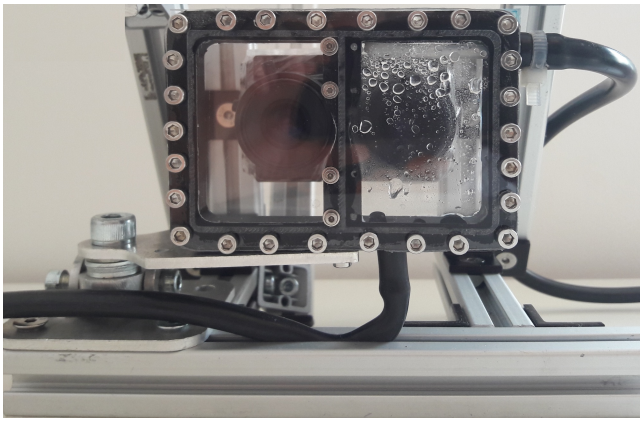


Fig. 8. Our small-baseline stereo camera setup. A bi-partite chamber with acrylic clear panels is placed in front of the lenses, with the left-hand section being kept dry at all times, while the right-hand section is sprayed with water droplets using an internal nozzle.

B. Training

We used a network training regimen similar to [30]. For each iteration we first trained the discriminator on a clear image and a de-rained image from a previous iteration with the goal of minimizing $\mathcal{L}_{\text{disc}}$, and then trained the generator on rainy input images to minimize $\mathcal{L}_{\text{gen}}$. We used the Adam solver [35] with an initial learning rate set at 0.0002, a batch size of 1, $\lambda_{\text{adv}} = 1$, $\lambda_{\text{perc}} = 1$ and $\lambda_{\text{msadv}} = 1$.

C. Segmentation Tasks

We used the trained generator G to de-rain all of the rainy input images. To benchmark both the images with computer-generated water drops and the images with real water drops, in the context of road marking segmentation, we used the approach of [4] which trains a U-Net to segment road markings in a binary way. To benchmark the computer-generated water drop images in the context of semantic segmentation, we used DeepLab v3 [36] which has achieved state-of-the-art performance on the Cityscapes dataset.

The generator runs at approximately 1 Hz for images with a resolution of 1280×960, and at approximately 3 Hz for images with a resolution of 640×480 on an Nvidia Titan X GPU.

V. Results

We benchmark our results taking into consideration several metrics across several tasks, and also present results on the quality of the image reconstruction.

A. Quantitative results

Table I presents results for road marking segmentation, in the case of RobotCar with real water drops (R), RobotCar with computer-generated water drops (S) and CamVid with computer-generated water drops (S). Our baseline is represented by the performance of clear images tested on models that were trained using clear images (REFERENCE). For both RobotCar (R), Robotcar (S), and the CamVid (S) datasets, the results show a severely degraded performance when testing rainy images on models that were trained using clear images (RAINY). Retraining the road marking segmentation models with a dataset augmented with rainy images will lead to an improvement in performance (AUGM). However, de-raining the images using our method and testing them on a model trained using clear images (DERAINED) restores the performance of the segmentation to levels that are close to the baseline recorded on clear images. Figure 4 shows road marking segmentation results on CamVid, before and after deraining. Figure 5 shows road marking segmentation results on RobotCar(R)&(S), before and after deraining.

As expected, re-training the segmentation model with a dataset that is augmented with rainy images helps to improve performance, however using a specialised de-raining preprocessing step significantly outperforms this approach, even when tested on a model trained exclusively with clear images. This is the expected advantage of having a model dedicated, in its entirety, to a specific image-to-image mapping task (de-raining), which narrows the variety of images fed to the segmentation task.

Table II presents results for semantic segmentation on the Cityscapes dataset. We benchmark 4 different combinations of models and datasets:

- Cityscapes-clear images tested on a model trained using Cityscapes-clear images;
- Cityscapes-rainy images tested on a model trained using Cityscapes-clear images;
- Cityscapes-rainy images tested on a model trained using Cityscapes-clear and Cityscapes-rainy images; and

TABLE I

ROAD MARKING SEGMENTATION RESULTS

Dataset	REFERENCE(CLEAR)				RAINY				AUGM.				DERAINED			
	Prec.	Rec.	F1	IOU	Prec.	Rec.	F1	IOU	Prec.	Rec.	F1	IOU	Prec.	Rec.	F1	IOU
RobotCar(R)	0.627	0.918	0.734	0.594	0.512	0.628	0.550	0.396	0.486	0.807	0.593	0.434	**0.603**	**0.841**	**0.689**	**0.544**
RobotCar(S)	0.627	0.918	0.734	0.594	0.364	0.595	0.437	0.287	0.654	0.770	0.690	0.541	**0.661**	**0.816**	**0.715**	**0.569**
CamVid(S)	0.576	0.927	0.699	0.551	0.353	0.576	0.425	0.279	0.457	**0.771**	0.563	0.405	**0.520**	0.755	**0.603**	**0.444**

TABLE II

CITYSCAPES SEMANTIC SEGMENTATION RESULTS

Cityscapes Img. vs. Segm. Model	mIOU
CLEAR on CLEAR	0.692
RAINY on CLEAR	0.405
RAINY on AUGMENTED	0.611
DERAINED on CLEAR	**0.651**

TABLE III

RECONSTRUCTION RESULTS

Dataset	RAW		DERAINED	
	PSNR	SSIM	PSNR	SSIM
RobotCar-Rainy(R)	13.02	0.5574	**22.82**	**0.8188**
RobotCar-Rainy(S)	16.80	0.6134	**25.17**	**0.8699**
CamVid-Rainy(S)	16.89	0.6064	**22.11**	**0.7524**
Qian et al.[11](R)	24.09	0.8518	**31.55**	**0.9020**

- Cityscapes-derained(Cityscapes-rainy preprocessed using our deraining model) images tested on a model trained using Cityscapes-clear images.

Similar to the case of road marking segmentation, we notice the same severe degradation of performance when testing with rainy images (RAINY on CLEAR) as compared to the baseline (CLEAR on CLEAR). Again, the performance of derained images tested on a model trained using clear images (DERAINED on CLEAR) is significantly better than the performance of rainy images tested on a model trained using a dataset augmented with rainy images (RAINY on AUGMENTED). Figure 6 shows semantic segmentation results on Cityscapes, before and after deraining.

B. Reconstruction results

Table III presents results on the quality of the image reconstruction using two widely used image-quality metrics, Peak signal-to-noise ratio (PSNR), and Structural similarity (SSIM). We benchmark our model on our real-world RobotCar-Rainy (R) dataset, RobotCar-Rainy with computer-generated rain (S), CamVid-Rainy with computer-generated rain (S), and on the dataset provided by [11]. The RAW column shows the quality of the rainy images, while the DERAINED column shows the quality of the de-rained images, all relative to their clear ground truth. We show that in all cases, de-raining the rain-affected images using our preprocessor significantly increases the quality of the images, as compared to the reference case where raw rainy images are used. Both the real-world rainy dataset images and the images with computer-generated rain are significantly more degraded than the rainy images provided by [11], as seen in column RAW.

Table IV presents reconstruction results on the reference rainy dataset provided by [11]. We show that we achieve state-of-the-art PSNR reconstruction results on images affected by real water drops and only slightly lower SSIM, while, in contrast to [11], not requiring an attention [28] mechanism, which simplifies and speeds up inference and training.

TABLE IV

RECONSTRUCTION QUALITY COMPARISON TO STATE OF THE ART

Model vs. Dataset	Dataset from [11]	
	PSNR	SSIM
Original	24.09	0.8518
Eigen13[21]	28.59	0.6726
Pix2Pix[37]	30.14	0.8299
Qian et al.(no att.)[11]	30.88	0.8670
Qian et al.(full att.)[11]	31.51	**0.9213**
Ours(no att.)	**31.55**	0.9020

VI. CONCLUSIONS

We have presented a system that restores performance of images affected by adherent raindrops on important segmentation tasks. Our results show that road marking segmentation, an important task for autonomous driving systems, is severely affected by adherent rain and that performance can be restored by first running the images through a de-raining preprocessor. Similarly, we show the same reduction and restoration of performance in the case of semantic segmentation, a task that is important in many fields. Additionally, we produce state-of-the-art results in terms of the quality of image restoration, while being able to run in real time. Finally, our system processes the image streams outside of the segmentation pipeline, either offline or online, and hence can be used naturally as a front end to many existing systems. The dataset will be made available at `https://ciumonk.github.io/RobotCar-rainy/`, along with a video describing our results at `https://ciumonk.github.io/RobotCar-rainy/video.html`.

VII. FUTURE WORK

Future work may involve designing a mechanism for producing computer-generated rain that is indistinguishable from real rain in terms of its usefulness in training models that quantitatively rather than qualitatively improve performance on image-based tasks.

VIII. ACKNOWLEDGEMENTS

This work was supported by Oxford-Google DeepMind Graduate Scholarships and Programme Grant EP/M019918/1. The authors wish to thank Valentina Musat for labelling the road markings in our dataset.

REFERENCES

[1] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3213–3223.

[2] E. Stenborg, C. Toft, and L. Hammarstrand, "Long-term visual localization using semantically segmented images," *CoRR*, vol. abs/1801.05269, 2018.

[3] J. L. Schönberger, M. Pollefeys, A. Geiger, and T. Sattler, "Semantic visual localization," *CoRR*, vol. abs/1712.05773, 2017.

[4] T. Bruls, W. Maddern, A. A. Morye, and P. Newman, "Mark yourself: Road marking segmentation via weakly-supervised annotations from multimodal data," in *Robotics and Automation (ICRA), 2018 IEEE International Conference on*. IEEE, 2018, p. in press.

[5] A. Valada, J. Vertens, A. Dhall, and W. Burgard, "Adapnet: Adaptive semantic segmentation in adverse environmental conditions," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, May 2017, pp. 4644–4651.

[6] Y. Chen, W. Li, C. Sakaridis, D. Dai, and L. V. Gool, "Domain adaptive faster R-CNN for object detection in the wild," *CoRR*, vol. abs/1803.03243, 2018.

[7] M. Wulfmeier, A. Bewley, and I. Posner, "Addressing appearance change in outdoor robotics with adversarial domain adaptation," *CoRR*, vol. abs/1703.01461, 2017.

[8] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[9] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 Year, 1000km: The Oxford RobotCar Dataset," *The International Journal of Robotics Research (IJRR)*, vol. 36, no. 1, pp. 3–15, 2017.

[10] G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognition Letters*, vol. 30, no. 2, pp. 88–97, 2009.

[11] R. Qian, R. T. Tan, W. Yang, J. Su, and J. Liu, "Attentive generative adversarial network for raindrop removal from a single image," *CoRR*, vol. abs/1711.10098, 2017.

[12] J. Chen and L. Chau, "A rain pixel recovery algorithm for videos with highly dynamic scenes," *IEEE Transactions on Image Processing*, vol. 23, no. 3, pp. 1097–1104, March 2014.

[13] J. Kim, J. Sim, and C. Kim, "Stereo video deraining and desnowing based on spatiotemporal frame warping," in *2014 IEEE International Conference on Image Processing (ICIP)*, Oct 2014, pp. 5432–5436.

[14] ——, "Video deraining and desnowing using temporal correlation and low-rank matrix completion," *IEEE Transactions on Image Processing*, vol. 24, no. 9, pp. 2658–2670, Sept 2015.

[15] W. Ren, S. Liu, H. Zhang, J. Pan, X. Cao, and M.-H. Yang, "Single image dehazing via multi-scale convolutional neural networks," in *European conference on computer vision*. Springer, 2016, pp. 154–169.

[16] X. Fu, J. Huang, X. Ding, Y. Liao, and J. Paisley, "Clearing the skies: A deep network architecture for single-image rain removal," *CoRR*, vol. abs/1609.02087, 2016.

[17] M. Roser and A. Geiger, "Video-based raindrop detection for improved image registration," in *2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops*, Sept 2009, pp. 570–577.

[18] M. Roser, J. Kurz, and A. Geiger, "Realistic modeling of water droplets for monocular adherent raindrop recognition using bézier curves," in *Computer Vision – ACCV 2010 Workshops*, R. Koch and F. Huang, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 235–244.

[19] S. You, R. T. Tan, R. Kawakami, Y. Mukaigawa, and K. Ikeuchi, "Waterdrop stereo," *CoRR*, vol. abs/1604.00730, 2016.

[20] J. F. Blinn, "A generalization of algebraic surface drawing," *ACM Trans. Graph.*, vol. 1, no. 3, pp. 235–256, July 1982.

[21] D. Eigen, D. Krishnan, and R. Fergus, "Restoring an image taken through a window covered with dirt or rain," in *2013 IEEE International Conference on Computer Vision*, Dec 2013, pp. 633–640.

[22] S. You, R. T. Tan, R. Kawakami, Y. Mukaigawa, and K. Ikeuchi, "Adherent raindrop modeling, detectionand removal in video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 9, pp. 1721–1733, Sept 2016.

[23] A. Yamashita, M. Kuramoto, T. Kaneko, and K. T. Miura, "A virtual wiper - restoration of deteriorated images by using multiple cameras," in *IROS*. IEEE, 2003, pp. 3126–3131.

[24] A. Yamashita, T. Kaneko, and K. T. Miura, "A virtual wiper-restoration of deteriorated images by using a pan-tilt camera," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, vol. 5, April 2004, pp. 4724–4729 Vol.5.

[25] A. Yamashita, Y. Tanaka, and T. Kaneko, "Removal of adherent waterdrops from images acquired with stereo camera," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Aug 2005, pp. 400–405.

[26] M. Kuramoto, A. Yamashita, T. Kaneko, and K. T. Miura, "Removal of adherent waterdrops in images by using multiple cameras," in *MVA*, 2002, pp. 80–83.

[27] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, ser. NIPS'14. Cambridge, MA, USA: MIT Press, 2014, pp. 2672–2680.

[28] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu, "Recurrent models of visual attention," *CoRR*, vol. abs/1406.6247, 2014.

[29] J. Cohen, M. Olano, and D. Manocha, "Appearance-preserving simplification," in *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '98. New York, NY, USA: ACM, 1998, pp. 115–122.

[30] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, A. Tao, J. Kautz, and B. Catanzaro, "High-resolution image synthesis and semantic manipulation with conditional GANs," in *Computer Vision and Pattern Recognition (CVPR), 2018 IEEE Conference on*. IEEE, 2018, pp. 1–13.

[31] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[32] C. Li and M. Wand, "Precomputed real-time texture synthesis with markovian generative adversarial networks," in *European Conference on Computer Vision*. Springer, 2016, pp. 702–716.

[33] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networkss," in *Computer Vision (ICCV), 2017 IEEE International Conference on*, 2017.

[34] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European Conference on Computer Vision*. Springer, 2016, pp. 694–711.

[35] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[36] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *ECCV*, 2018.

[37] P. Isola, J. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," *CoRR*, vol. abs/1611.07004, 2016.

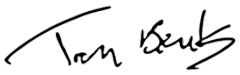
Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor **(only required where there isn't already a statement of contribution within the paper itself).**

Title of Paper	I Can See Clearly Now: Image Restoration via De-Raining
Publication Status	Published
Publication Details	H. Porav, **T. Bruls**, and P. Newman, "I can see clearly now: Image restoration via de-raining", in *Proceedings of the International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 7087-7093.

Student Confirmation

Student Name:	Tom Adriaan Hubert Bruls		
Contribution to the Paper	Contributions included: - Refining the initial ideas. - Preparing the datasets for road marking segmentation. - Supporting the execution of the road marking segmentation experiments. - Evaluation of the road marking segmentation experiments. - General editing of the paper. The overall paper emerged as a product of discussions and collaboration with my co-authors.		
Signature		Date	10-05-2020

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Professor Paul Newman		
Supervisor comments		
Signature		Date 11-05-2020

This completed form should be included in the thesis, at the end of the relevant chapter.

C

Semantic Classification of Road Markings from Geometric Primitives

This appendix contains a reproduction of the following publication:

> [23] P. Amayo, **T. Bruls**, and P. Newman, "Semantic classification of road markings from geometric primitives", in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Nov. 2018, pp. 387-393.

Semantic Classification of Road Markings from Geometric Primitives

Paul Amayo, Tom Bruls, and Paul Newman

***Abstract*— The classification of semantically meaningful road markings in images is an important and safety critical task for autonomous and semi-autonomous vehicles. However, beyond simple lane markings, real-time detection and interpretation of road markings is challenging as images are subject to occlusions, partial observations, lighting changes and differing weather conditions. Additionally, there is high variation in the road markings between countries and regions, which makes interpretation difficult. In this work we present a three-fold approach to the semantic classification. Firstly, we employ a weakly supervised neural network to detect pixels belonging to road markings under different conditions. Subsequently, these pixels are classified into geometric primitives, from which we retrieve the semantic classes through a fast and parallel model-fitting algorithm that offers real-time performance. Unlike other methods in the literature that perform road marking classification independently, our proposed approach performs a joint classification leveraging the highly structured configurations that characterise urban traffic scenes. Consequently, we retrieve the underlying semantic classes under a variety of weather and lighting conditions as we demonstrate in our results.**

I. Introduction

The safe operation and deployment of a robot is intrinsically tied to its understanding of the work-space it operates in. In the case of an autonomous vehicles this extends from the knowledge of its location and its surroundings to the allowable behaviour at that particular location. The latter is mainly encoded into painted markings on the road surface which guide vehicles into acceptable behaviour and serve as warning for different hazards.

Offline mapping services such as Google Maps, HERE maps, and OpenStreetMap nowadays attempt to include these kind of details to aid autonomous driving. However, these offline systems do not fully negate the need for autonomous vehicles to be able to directly detect and interpret road markings in real-time through their live sensors for several reasons (i.e. roads are constantly changing, increasing, or undergoing maintenance). These off-line methods cannot directly compensate for this, leading to safety concerns for autonomous operation especially as regions that receive less traffic are considered.

Therefore, we focus on scene understanding for autonomous vehicles from live perception. More specifically, we present a method for the classification of a collection of road markings (i.e. not just lane markings) from a front-facing monocular camera. We operate on the premise that the majority of the painted road markings originate from simple geometric primitives (i.e. lines), even though their scale and rotation differ greatly due to the camera perspective. The spatial configuration of these primitives distinguishes the road marking classes.

The authors are with the Oxford Robotics Institute, Dept. Engineering Science, University of Oxford, UK. {pamayo, tombruls, pnewman}@robots.ox.ac.uk

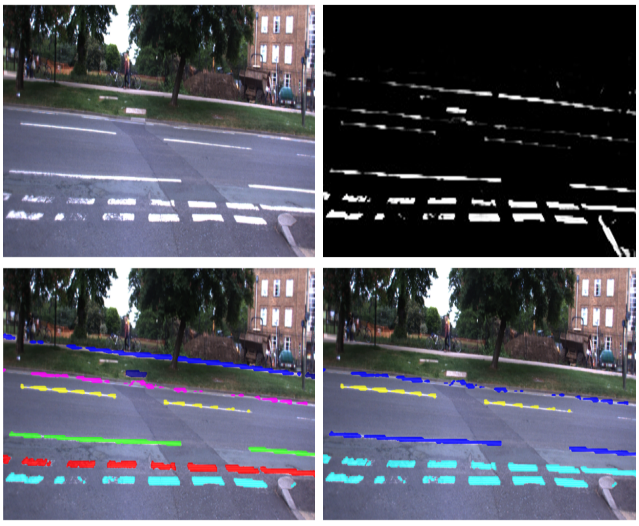


Fig. 1. Semantic classification of road markings in an urban environment as implemented in this paper. From an image taken by the front-facing camera of an autonomous vehicle (*top left*), pixels of potential road markings can be identified by a trained deep neural network (*top right*) under various lighting conditions. A fast and real-time two-step global energy optimisation approach then retrieves the road marking classes. The first optimisation step reveals geometric primitives (lines) which encode road marking segments (*bottom left*), before a further optimisation step classifies these primitives into semantically meaningful road markings as seen in the *bottom right* image.

Early approaches for road marking classification [1], [2] proposed the matching of features/shapes to obtain the road marking classes from geometric primitives, but generally struggle with changes in orientation and scale. Additionally, these approaches performed shape classification independently, neglecting the fact that combinations of road markings are often found in the same spatial configuration. Furthermore, there is a decrease of performances in scenes with occlusions or changes in lighting and weather conditions. End-to-end deep learning approaches [3] offer solutions to some of these problems. However, creating pixel-wise road marking classification labels is extremely labour expensive and cannot be done automatically as in [4]. Besides, no techniques currently exist for including domain knowledge (e.g. structure of the road scene or geometric primitives of road markings) directly into these deep learning frameworks.

In this paper, we propose an integrated framework for the detection and semantic classification of road markings. As demonstrated in Figure 1, pixels that belong to road markings are first identified by a trained deep neural network [4].

This allows for accurate detection of road marking pixels under varying lighting and weather conditions. From these pixels a two-step method retrieves the semantic road marking classes. This method is based on a fast, robust, real-time, global energy optimisation implemented through a COnvex Relaxation Algorithm (CORAL) [5]. Unlike most geometric multi-model fitting approaches, global energy approaches inherently consider the overall classification of data points to underlying models, in this case considering the spatial proximity of road marking classes. In the first optimisation step, geometric primitives (i.e. lines) from the identified pixels are extracted. These are then clustered through a subsequent optimisation, due to their specific configurations, to the different semantic classes through a further energy optimisation revealing a joint classification of road markings in a scene as shown in the bottom right of Figure 1. This removes the need for expensive and time-consuming manual annotations used for training purposes in proposed learning techniques [3], while still retaining strong and real-time classification performance.

In particular this work offers the following contributions:

- A robust, accurate method for extracting geometric primitives from road marking pixels.
- A fast, global labelling for the semantic classification of road markings using a combination of the geometric primitives and the road marking pixels.
- A method to track road marking classes from frame to frame.

The rest of the paper is organised as follows. In Section II related work in the area of road marking detection is summarised. In Section III the process for obtaining the road marking pixels is explained, followed by Section III-B wherein the details of the geometric primitive extraction technique are described, before presenting the subsequent segmentation in Section III-C. In Section IV we introduce the road marking tracking pipeline before showing qualitative and quantitative results in Section V. Conclusions and discussion follow in Section VI.

II. Related work

Early work on the classification and interpretation of road markings was majorly concerned with the detection of lane markings, which form a subset of the road markings but are important for semi-autonomous vehicles as lane-following and lane departure warning systems are popular Advanced Driver Assistance Systems (ADAS). The survey by Hillel et al. [6] presents the common pipeline implemented by most lane marking detection algorithms using images. This pipeline includes a feature detection step to extract the outlines of the lane separators through edge or gradient detection, followed by a model fitting algorithm, usually a Hough transform for straight roads and a spline or polynomial model for curved roads. However, the feature detection is prone to occlusions and changes in illumination and weather conditions, while the model-fitting is dependent on correct parametrization and model choice. This limits the usability of these techniques to simple scenes (e.g. highway environments), not alike the urban environments studied in this paper.

To improve on the aforementioned problems, several approaches introduced an extra filtering step that attempts to extract regions of the viewed scene which contain road markings, as presented in the survey by Veit et al. [7]. While these are able to improve the baseline performance, they still suffer under varying illumination forcing the use of additional heuristics for practical functionality [8], [9]. Additionally as more complex road markings are sought the extracted features evolve from simple edges to retrieve lanes to more complex shape contours to detect arrows and spatial configurations for zebra crossings. Examples of these are descriptor-based approaches such as Histogram of Gradients (HOG) used in [10] and Fourier approaches used in [11], [12]. Classification of these features can then be performed by template matching as seen in [13], [14] or by heuristic shape-based rules. These techniques, while showing good performance on their evaluated datasets, do not generalise well and are sensitive to occlusions and partial observations limiting their practical use.

Supervised learning approaches offered a way to improve generalisation and thus several flavours of these appeared in the literature, ranging from KNN classification [15] and Support Vector Machines [16], [17] to shallow neural networks [18], [19]. It must be noted that these techniques each focus on a different specific subset of road markings, which are then independently detected and classified and hence generalisation to new features is limited. A divergence from this approach is presented by Bonolo et al. [20], who exploited the spatial relationship between road markings to improve classification using a Conditional Random Field (CRF). This however requires perfect detection of road markings and with the optimisation taking a few seconds per image it is not suited for online use.

More recently, deep networks have been successfully trained for road marking recognition [21], [22] or purely for classification [23]. However, these approaches either implement additional preprocessing algorithms or require detected road markings as an input, because of a lack of ground-truth road marking labels in large-scale urban datasets. The authors of [3] are the first to train a network on a large-scale (hand-labelled) dataset and perform coarse road marking detection under challenging conditions.

In comparison to all of the aforementioned work, our approach leverages the power of deep learning to perform robust pixel-accurate road marking detection under difficult conditions and occlusions without the need for expensive road marking class labels, while still integrating domain knowledge to semantically classify road markings jointly in real time.

III. System Overview

In this work, we take a three-fold approach to the semantic classification of road markings. Firstly, we deploy a deep semantic segmentation network to detect road marking pixels $\mathbf{u} \in \{u_1, \cdots, u_{n_p}\}$, where n_p is the number of detected pixels, in a monocular image. This non-trivial operation removes artefacts from the image such as cars, buildings, and other objects with line features, which introduce noise to the subsequent steps.

Secondly, a global energy optimisation retrieves an *a priori* unknown number of geometric primitives $A \in \{A_1, \cdots, A_{n_l}\}$, where n_l is the number of extracted primitives, from the road marking pixels which encode the road marking segments.

Lastly, we cluster these road marking segments into semantically meaningful classes $\psi \in \{\psi_1, \cdots, \psi_{n_c}\}$, with two types of constraints followed by a subsequent energy optimisation. This pipeline is described in more detail in the following subsections.

A. Road Marking Detection

We deploy a deep semantic segmentation network to identify image pixels that belong to the road markings. It can be seen [4] that in the presence of adequate training data, such a network will outperform existing techniques as it is able to exploit the global scene context, making it more robust to lighting changes, spatial deformations, degradation, and partial occlusions.

Creating adequate training data, in this case pixel-wise road marking labels, is extremely labour expensive. To bootstrap this, the monocular image is combined with a LiDAR reflectance point cloud to create road marking annotations in a weakly supervised way using several domain assumptions. We exploit the property that road markings are highly reflective and optimise a dense CRF over the image to detect the road marking pixels by relating them to high-reflectance LiDAR points, which are not affected by lighting changes. This allows for the automatic generation of a large set of road marking annotations under various conditions, which are used for training purposes.

After the network is trained with these annotations, a road marking mask can be retrieved in real-time from a monocular image. Figure 2 shows that the network is robust to both appearance and lighting changes in the scene, providing a strong set of pixels from which individual road markings can be obtained. Due to space constraints we direct the reader to [4] for further details and the utilised network architecture.

B. Road Marking Geometric Primitive Extraction

After the identification of the road marking pixels by the trained deep segmentation network, we extract geometric primitives. linear segments, that encapsulate road marking segments. The number of road marking segments in a specific scene is *a priori* unknown, and apart from the trivial case of lane markings, they occur in different orientations and lengths with various levels of occlusion. Additionally, the network output also contains some level of noise, as objects of high reflectance (i.e. curbs) not corresponding to road markings can sometimes be wrongly segmented.

Several techniques for multi-line fitting in images such as vanishing point detection [3], RANSAC [24] and the Hough transform have been employed to obtain road marking segments. However, when moving away from the simple lane detection scenario performance of these techniques deteriorates. The Hough transform is highly dependent on parametrisation, while sequential RANSAC techniques struggle in scenes with noise and clutter, which cascades inaccuracy leading to poor outputs. In contrast, robust energy based

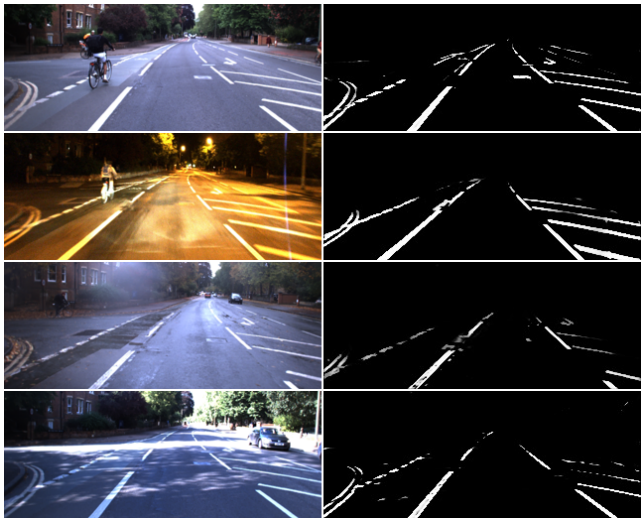


Fig. 2. Road marking detection under different lighting conditions (overcast, night, rainy, and sunny). Despite the large changes in the prevailing conditions the trained deep segmentation network [4] is able to accurately identify the pixels belonging to the road markings from the monocular image.

multi-model fitting approaches [25], [5] have been shown to outperform greedy approaches in the presence of noise. This is as energy based approaches take into consideration the overall classification of the data points to all the models. Firstly, by promoting locality through a smoothness prior that ensures that points that are close together have a similar model. In addition, these techniques are able to converge to the correct number of geometric models present in the data through a compactness prior.

With this in mind, we adopt the formulation given by CORAL [5] to perform the multi-line fitting. We define a global energy function:

$$\underbrace{\sum_{l=1}^{n_l}\sum_{i=1}^{n_{rm}}(\|D(\mathbf{A}_l,\mathbf{u}_i)\|)\phi_l(\mathbf{u})}_{\text{Geometric Error Energy}} + \lambda \underbrace{\sum_{l=1}^{n_l}\sum_{i=1}^{n_{rm}}|\nabla_{\mathcal{N}}\phi_l(\mathbf{u})|_1}_{\text{Smoothness Energy}} + \underbrace{\beta\|L\|}_{\text{Compactness Energy}} \tag{1}$$

The data term in Equation 1 accounts for the distance between a point and the geometric primitive. Here A is the line equation $\mathbf{A}_l = (a_l, b_l, c_l)$ and we refer to D as the Euclidean distance between a point $\mathbf{u}_i = (x_i, y_i)$ and the line A_l. Membership of data points to their respective model is encapsulated through the indicator function

$$\phi_l(\mathbf{u}) = \begin{cases} 1 & \mathbf{u} \in A_l \\ 0 & \text{otherwise} \end{cases}, \tag{2}$$

which is self-constrained, $\sum_{l=1}^{n_l}\phi_l(\mathbf{u}) = 1$, such that a point can only be a member to one model. To account for outliers – as not all data points might be explained by a linear segment– a special label $\emptyset$, representing the outlier model is added. In this way a constant cost γ is assigned to points that cannot be explained by any geometric model.

Fig. 3. A sample of the results from road marking geometric primitive extraction under different prevailing conditions. It can be seen that our proposed energy optimisation approach is able to accurately extract linear segments in a diversity of scenes. Revealing the underlying primitives for a large number of road markings present in the urban scene.

The smoothness term in Equation 1 promotes a homogeneous assignment of models to neighbouring points, introducing a spatial smoothness prior. By calculating the gradient $\nabla_{\mathcal{N}}$ of the indicator function over the neighbourhood $\mathcal{N}$ of a point given by its k-nearest neighbours, points that belong to the same neighbourhood but do not share the same model are penalised. The trade-off between the smoothness and data terms is controlled by the parameter λ. Finally, the compactness term in Equation 1 penalises the number of models by adding a constant cost β per model. This penalises redundant models resulting in a more compact solution.

Minimisation of the energy in Equation 1 reveals the underlying geometric models. In CORAL a continuous optimisation approach leveraging a primal dual optimisation [26] is employed. This approach is inherently parallelisable allowing for easy implementation on General Purpose Graphical Processing Unit (GPGPU) hardware and real-time line model detection. Due to space constraints we refer the reader to [5] for further implementation details.

To reduce the search space for models in the CORAL optimisation, a finite number of models is usually proposed. In this work we use the Hough Transform for the model initialisation which generally proposes more models with low accuracy than are present in the scene. This is followed by an iterative process of primal dual optimisation for energy optimisation and model re-estimation up until the energy converges. Thus converging on n_l road marking segments. A sample of the results can be seen in Figure 3, where this approach is able to extract the underlying primitives for a large number of road markings in a diversity of scenes.

C. Road Marking Geometric Primitive Clustering

The approach described in the previous section is able to accurately extract the road marking geometric primitives under different conditions. However, it is the underlying meaning encoded in these primitives that is actually interesting for autonomous driving. Therefore, we seek a clustering of the geometric primitives to perform classification. For the clustering, we utilise the idea that a collection of geometric primitives, however complicated, is still a geometric model albeit with more intra-class constraints $\theta(\cdot)$. Additionally, a set of fixed rules governs the spatial relationships between road marking classes which can be encoded into inter-class constraints $\Omega(\cdot)$.

Before the clustering can be performed, the effect of the camera perspective must be removed as it distorts the length, orientation and position of the road marking segments. By positioning a virtual camera above the observed scene this effect can be removed providing consistency not only in the detected road marking segments but also in their spatial configurations. This is performed through a homography warping of the scene, referred to as the Inverse Perspective Mapping (IPM) [20], which despite assuming that the road surface is planar works well in practice as seen in Figure 4.

We focus on the detection of six classes of road markings: single lane boundaries, double lane boundaries, lane separators, intersection markings, zig-zag, and junction road markings. Detection of these classes informs an autonomous vehicle about an upcoming road situation (e.g. a pedestrian crossing or a junction) or the allowable drivable area, both are crucial functions for autonomous driving.

Double lane boundaries, zig-zag and junction road markings all originate from a collection of geometric primitives and thus introduce intra-class constraints. These constraints are the angle and the distance between two road marking segments. Given the equation of a geometric primitive $\mathbf{A}_l = (a_l, b_l, c_l)$, we introduce two simple constraints as follows

With these classes in mind two constraints were introduced, the angle between road marking segments and the corresponding distance between them. While simple these constraints encompass the possible configurations of the road marking classes and can be defined as:

$$\theta_{angle}(\mathbf{A}_i, \mathbf{A}_j) = |\arctan(a_i/b_i) - \arctan(a_j/b_j)| \quad (3)$$

$$\theta_{dist}(\mathbf{A}_i, \mathbf{A}_j, \mathbf{u}_i^m, \mathbf{u}_j^m) = (D(\mathbf{A}_l, \mathbf{u}_i^m) + D(\mathbf{A}_l, \mathbf{u}_j^m))/2, \quad (4)$$

where $\mathbf{A}_i = (a_i, b_i, c_i)$ is the equation of the road marking segment i and $\mathbf{u}_i^m$ is the midpoint of the pixels assigned to it.

By observing the classes it can be seen that the lane boundaries, lane separators and intersection road markings all consist of singular infinite line models. There is however, a strong prior that these classes appear parallel to each other allowing for the definition of an inter-class constraint that

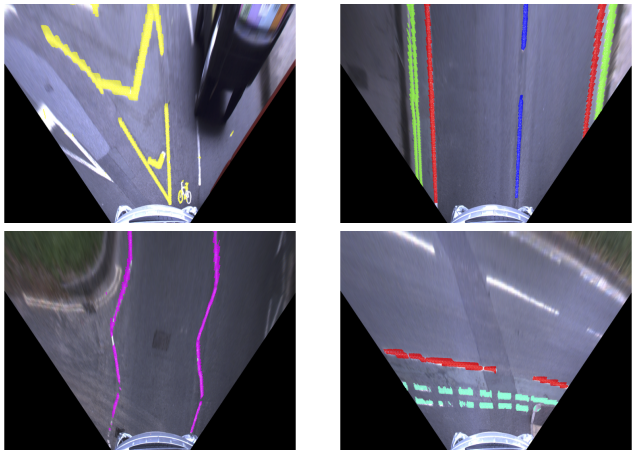


Fig. 4. Manually annotated road markings after inverse perspective mapping. These show a junction (*top left*), lane markings (*top right*), zig-zag lines (*bottom left*), and a road intersection (*bottom right*) scene. As the semantic classes are encoded through a specific configuration of geometric primitives a subsequent CORAL optimisation reveals the underlying road marking classes.

penalises a collection of these if they are not parallel through the angle constraint. Similarly the double lane boundary consists of two parallel singular infinite line models within close proximity of each other, providing a inter-class constraints based on the angle and distance. Lastly, the angle is preserved in the zig-zag and junction crossings classes, producing a similar inter-class constraint.

We can thus propose several instances of these semantic classes through a targeted search that aims to find collection of lines that are parallel as well as those that fulfil the angle constraints of the zig-zag and junction classes. These instances form an initial set of "models" that are fed into the CORAL. The iterative optimisation of which reveals a compact set of semantic class instances as but additionally optimises for spatial smoothness, in essence performing a *joint* optimisation.

A sample of results of these is given in Figure 5, showing that this approach achieves high classification accuracy even in the presence of occlusions. To differentiate between the three classes of singular infinite line models in these results, the contiguity of their associated pixel inliers is used.

IV. Road Marking Tracking

The previous section has described a framework for retrieving the road marking classes in a single image frame. Tracking of the road markings through consecutive frames improves the robustness of the classification, because, in most cases, road markings are seldom only seen in one frame, and persist from frame to frame. By exploiting this persistence our confidence of correct classification is increased when a road marking is detected over multiple frames. Additionally, some road markings are not fully observable in the current image (e.g. when traversing a road junction), making their classification ambiguous. By tracking classes, we can ensure that the correct assignment is made even when the road markings become partially observed.

Algorithm 1: Multi-Frame Road Marking Tracking

```
if First Frame then
    Propose Θ̂_0 models with HT;
else
    Calculate homography;
    Warp lines Θ_{i-1} to Θ̃_i;
    Remove inliers;
    Propose Θ_HT models from outliers using HT;
    Θ̂_i = {Θ̃_i, Θ_HT};
end
while not converged do
    Primal Dual Optimisation;
    Merge lines;
    Re-estimate lines;
end
N road marking segments Θ_i;
Semantic road markings;
```

In this work, the selected road marking classes are collections of geometric primitives. These primitives can be tracked between subsequent frames if the motion $\mathbf{T} = \{\mathbf{R}, \mathbf{t}\}$ between the frames is known. We use the homography transformation between the subsequent frames to project a line model into the next frame

$$\mathbf{H} = \mathbf{R} + \frac{\mathbf{t}\mathbf{n}^t_{plane}}{d_{plane}}$$
$$\mathbf{A}_{i+1} = \mathbf{H}\mathbf{A}_i. \tag{5}$$

These give an initial set of projected models for the subsequent frame, obtained from the initial frame. However, this set does not include all possible models, as road marking segments can appear for the first time in a particular frame. To cope with this, road marking pixels that are inliers to the projected models are first removed before a further Hough Transform (HT) initialisation is performed to the outliers availing new road marking geometric primitives as summarised in Algorithm 1. We implement a sliding-window approach to track road marking classes when they become fully occluded or the image becomes over-exposed. This allows detected semantic classes to persist further in time more accurately.

V. Experimental Results

To evaluate the presented approach, the Oxford RobotCar dataset [27] is used. This dataset consists of 100 repetitions of a 10-km route in central Oxford under different prevailing weather and lighting conditions. Using the pre-trained deep segmentation network [4], we deploy our approach on three runs that were captured under vastly different conditions (overcast, rain, and night-time). It can be seen that our approach is able to retrieve the underlying semantic classes of the road markings in a variety of scenes even despite significant changes in appearance, as shown in Figure 5.

In Figure 6, we demonstrate the benefit of our road marking tracking algorithm. In this sequence, the junction markings become partially observable, leading to wrong

Fig. 5. Sample of results from the semantic classification of road marking pixels under different weather and lighting conditions (overcast/rain/night). By using a global energy approach, our proposed approach is able to detect multiple road marking segments in images from detected road marking pixels. These segments are aggregated through another energy minimisation into their semantic classes. Thereby, we reveal the underlying meaning of the road markings in complex urban environments, providing important cues for autonomous vehicles. These include indication for upcoming road situation, which could require specific behaviour. For instance, the zig-zag markers (*purple*) indicate an upcoming pedestrian crossing and the give-way dashes (*cyan*) indicate a junction.

Fig. 6. Given the fully observed junction road marking (*left*), our approach is able to correctly detect the underlying semantic class. However, when the junction becomes partially observed (*middle*), the interaction of the underlying geometric primitives with others in the scene can cause mis-classification. In this case, the junction was labelled as a zig-zag line. By introducing road marking tracking (*right*), the correct class is retrieved even under partial observation, making the classification more robust.

classification when only the current image frame is taken into account. By tracking the road markings, our approach memorises the scene and correctly initialises a new semantic road marking, increasing the robustness of the system.

The CORAL global energy optimisation is implemented using CUDA and deployed on an NVIDIA TITAN GPU. To obtain the running times, we averaged the computational time of 100 different images from the dataset. The timing results are presented in Table I. The results show that this method can be performed in real time ($\sim$6 Hz), allowing online road marking classification.

TABLE I
TIMING RESULTS FOR THE ROAD MARKING SEGMENTATION

Module	Time (ms)
Road marking pixel detection	16
Energy minimisation (line models)	110.8
Energy minimisation (semantic classes)	38.4

VI. CONCLUSION

In this paper we have presented a framework for the classification and interpretation of road markings in complex urban environments under varying weather conditions. From detected road marking pixels, this approach describes the semantic classes as different configurations of primitive geometric models and then employs a fast energy minimisation to extract the respective class in real time. By detecting certain classes we are able to reason about upcoming road situations, which could require specific behaviour. Unlike most of the contemporary approaches, we classify road markings jointly without requiring expensive manual annotations and are able to perform well in the presences of occlusions and degradation. Furthermore, the method is easily extendable to more classes and is thus able to provide an important cue for planning and navigation in urban scenes.

VII. ACKNOWLEDGEMENTS

The authors acknowledge the following funding sources. Paul Amayo is funded by the Rhodes Trust. Paul Newman is supported by EPSRC Programme Grant EP/M019918/1.

REFERENCES

[1] G. Maier, S. Pangerl, and A. Schindler, "Real-time detection and classification of arrow markings using curve-based prototype fitting," in *Intelligent Vehicles Symposium (IV), 2011 IEEE*. IEEE, 2011, pp. 442–447.

[2] X. Du and K. K. Tan, "Comprehensive and practical vision system for self-driving vehicle lane-level localization," *IEEE transactions on image processing*, vol. 25, no. 5, pp. 2075–2088, 2016.

[3] S. Lee, J. Kim, J. S. Yoon, S. Shin, O. Bailo, N. Kim, T.-H. Lee, H. S. Hong, S.-H. Han, and I. S. Kweon, "Vpgnet: Vanishing point guided network for lane and road marking detection and recognition," in *Computer Vision (ICCV), 2017 IEEE International Conference on*. IEEE, 2017, pp. 1965–1973.

[4] T. Bruls, W. Maddern, A. A. Morye, and P. Newman, "Mark yourself: Road marking segmentation via weakly-supervised annotations from multimodal data," in *Robotics and Automation (ICRA), 2018 IEEE International Conference on*. IEEE, 2018, p. in press.

[5] P. Amayo, P. Piniés, L. M. Paz, and P. Newman, "Geometric Multi-Model Fitting with a Convex Relaxation Algorithm," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, Salt Lake City, USA, June 2018.

[6] A. B. Hillel, R. Lerner, D. Levi, and G. Raz, "Recent progress in road and lane detection: a survey," *Machine vision and applications*, vol. 25, no. 3, pp. 727–745, 2014.

[7] T. Veit, J.-P. Tarel, P. Nicolle, and P. Charbonnier, "Evaluation of road marking feature extraction," in *Intelligent Transportation Systems, 2008. ITSC 2008. 11th International IEEE Conference on*. IEEE, 2008, pp. 174–181.

[8] R. Danescu and S. Nedevschi, "Detection and classification of painted road objects for intersection assistance applications," in *Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on*. IEEE, 2010, pp. 433–438.

[9] Z. Liu, S. Wang, and X. Ding, "Roi perspective transform based road marking detection and recognition," in *Audio, Language and Image Processing (ICALIP), 2012 International Conference on*. IEEE, 2012, pp. 841–846.

[10] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

[11] J. M. Collado, C. Hilario, A. de la Escalera, and J. M. Armingol, "Detection and classification of road lanes with a frequency analysis," in *Intelligent Vehicles Symposium, 2005. Proceedings. IEEE*. IEEE, 2005, pp. 78–83.

[12] J. M. Collado, C. Hilario, A. De La Escalera, and J. M. Armingol, "Adaptative road lanes detection and classification," in *International Conference on Advanced Concepts for Intelligent Vision Systems*. Springer, 2006, pp. 1151–1162.

[13] P. Foucher, Y. Sebsadji, J.-P. Tarel, P. Charbonnier, and P. Nicolle, "Detection and recognition of urban road markings using images," in *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on*. IEEE, 2011, pp. 1747–1752.

[14] S. Vacek, C. Schimmel, and R. Dillmann, "Road-marking analysis for autonomous vehicle guidance." in *EMCR*, 2007.

[15] J. Rebut, A. Bensrhair, and G. Toulminet, "Image segmentation and pattern recognition for road marking analysis," in *Industrial Electronics, 2004 IEEE International Symposium on*, vol. 1. IEEE, 2004, pp. 727–732.

[16] L. Gang, M. Zhang, L. Zhang, and J. Hu, "Automatic road marking recognition for intelligent vehicle systems application," *Advances in Mechanical Engineering*, vol. 9, no. 5, p. 1687814017706267, 2017.

[17] D. Ding, J. Yoo, J. Jekyo, S. Jin, and S. Kwon, "Efficient road-sign detection based on machine learning," *Bulletin of Networking, Computing, Systems, and Software*, vol. 4, no. 1, pp. 15–17, 2015.

[18] A. Kheyrollahi and T. P. Breckon, "Automatic real-time road marking recognition using a feature driven approach," *Machine Vision and Applications*, vol. 23, no. 1, pp. 123–133, 2012.

[19] J. Yamamoto, S. Karungaru, and K. Terada, "Road surface marking recognition using neural network," in *System Integration (SII), 2014 IEEE/SICE International Symposium on*. IEEE, 2014, pp. 484–489.

[20] B. Mathibela, P. Newman, and I. Posner, "Reading the road: Road marking classification and interpretation," *IEEE Trans. Intelligent Transportation Systems*, vol. 16, no. 4, pp. 2072–2081, 2015. [Online]. Available: http://dx.doi.org/10.1109/TITS.2015.2393715

[21] O. Bailo, S. Lee, F. Rameau, J. S. Yoon, and I. S. Kweon, "Robust road marking detection and recognition using density-based grouping and machine learning techniques," in *Applications of Computer Vision (WACV), 2017 IEEE Winter Conference on*. IEEE, 2017, pp. 760–768.

[22] T. Chen, Z. Chen, Q. Shi, and X. Huang, "Road marking detection and classification using machine learning algorithms," in *Intelligent Vehicles Symposium (IV), 2015 IEEE*. IEEE, 2015, pp. 617–621.

[23] T. Ahmad, D. Ilstrup, E. Emami, and G. Bebis, "Symbolic road marking recognition using convolutional neural networks," in *Intelligent Vehicles Symposium (IV), 2017 IEEE*. IEEE, 2017, pp. 1428–1433.

[24] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[25] H. Isack and Y. Boykov, "Energy-based geometric multi-model fitting," *International journal of computer vision*, vol. 97, no. 2, pp. 123–147, 2012.

[26] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *Journal of Mathematical Imaging and Vision*, vol. 40, no. 1, pp. 120–145, 2011.

[27] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000 km: The oxford robotcar dataset," *The International Journal of Robotics Research*, vol. 36, no. 1, pp. 3–15, 2017. [Online]. Available: http://dx.doi.org/10.1177/0278364916679498

Statement of Authorship for joint/multi-authored papers for PGR thesis

To appear at the end of each thesis chapter submitted as an article/paper

The statement shall describe the candidate's and co-authors' independent research contributions in the thesis publications. For each publication there should exist a complete statement that is to be filled out and signed by the candidate and supervisor **(only required where there isn't already a statement of contribution within the paper itself).**

Title of Paper	Semantic Classification of Road Markings from Geometric Primitives
Publication Status	Published
Publication Details	P. Amayo, **T. Bruls**, and P. Newman, "Semantic classification of road markings from geometric primitives", in *Proceedings of the Intelligent Transportation Systems Conference (ITSC)*, Nov. 2018, pp. 387-393.

Student Confirmation

Student Name:	Tom Adriaan Hubert Bruls		
Contribution to the Paper	Contributions included: - Refining the initial ideas. - Preparing the dataset for road marking segmentation. - Running the road marking segmentation experiments. - Co-writing the introduction, related work, and road marking detection section. - Creating the road marking detection figure. - Presenting the work at the conference. The overall paper emerged as a product of discussions and collaboration with my co-authors.		
Signature		Date	10-05-2020

Supervisor Confirmation

By signing the Statement of Authorship, you are certifying that the candidate made a substantial contribution to the publication, and that the description described above is accurate.

Supervisor name and title: Professor Paul Newman			
Supervisor comments			
Signature		Date	11-05-2020

This completed form should be included in the thesis, at the end of the relevant chapter.