2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
Madrid, Spain, October 1-5, 2018

Deep Semantic Lane Segmentation for Mapless Driving

Annika Meyer[1], N. Ole Salscheider[1], Piotr F. Orzechowski[1] and Christoph Stiller[2]

***Abstract*— In autonomous driving systems a strong relation to highly accurate maps is taken to be inevitable, although street scenes change frequently. However, a preferable system would be to equip the automated cars with a sensor system that is able to navigate urban scenarios without an accurate map. We present a novel pipeline using a deep neural network to detect lane semantics and topology given RGB images. On the basis of this classification, the information about the road scene can be extracted just from the sensor setup supporting mapless autonomous driving. In addition to superseding the huge effort of creating and maintaining highly accurate maps, our system reduces the need for precise localization.**

Using an extended Cityscapes dataset, we show accurate ego lane detection including lane semantics on challenging scenarios for autonomous driving.

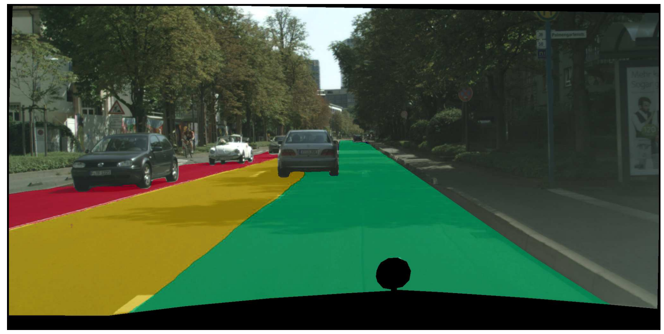

Fig. 1. An example result from the Cityscape's Frankfurt dataset segmented into its semantic lanes. Shadows and vanishing markings on the street make the detection of the ego lane (green) and parallel traffic (yellow) difficult, but our approach is able to estimate the lane boundaries and semantic correctly. It also correctly determines the oncoming traffic (red). The extracted semantic information is crucial for planning the car's trajectory and steer it safely, when not relying on a map.

I. INTRODUCTION

The lane structure is a crucial information for safe autonomous driving especially in urban environments. An autonomous car must identify the current lane robustly in order to steer the vehicle safely.

Current autonomous driving systems, such as BerthaONE [1], often resort to an accurate and detailed map of the environment to tackle this issue. However, these maps need to be updated constantly due to road constructions and can therefore be quickly out of date. Furthermore, a map-based approach requires a precise localization of the vehicle, that is typically weak in narrow urban scenes, due to multi-path effects.

A. *Feature-based Geometry Estimation*

An early approach for estimating the road geometry uses visual features and object detections [2]. The authors present a real-time approach using both spatial and temporal aspects to estimate road curvature parameters. Many later approaches for estimating the road layout also rely on distinct features like lane markings and curbs [3]–[6].

Beck et al. [7] increased the robustness of the markings and curb based approaches by incorporating vanishing points and a free-space estimation. In [8] previous work is extended with a coarse semantic segmentation for the lane estimation process. Another possible solution is to extend the road boundary information by the position of other vehicles and to reason about relations between them in order to make an assumption on the drivable path [9].

The extracted features are treated differently depending on the approach and its goal. The most common algorithms fit a model to the markings and curbs using a hyperbola [4] or a clothoid ([9], [10]). More abstract model fitting is presented in [3] and [7], where the road is estimated using a graphical model that is based on visual features. Similar to that, [11] apply a probabilistic framework in order to represent the uncertainty of the measurements. In [12], those approaches are extended by including rough map priors. In 2018 [13] proposed an approach, where the lane boundaries are estimated using instance segmentation based on the road markings.

The main drawback of these approaches is a strong dependency on road markings or curbs. However, especially in urban scenarios markings might either be worn out, not be existing or occluded [11]. Hence, a system relying only on handcrafted features is not suitable for autonomous driving in urban environments, since it also must deal with missing or erroneous data. For this reason, we vote for avoiding purely visual, handcrafted features and instead for training a network to learn the relevant information itself.

Caltagirone et al. [14] replace handcrafted features with a deep neural network, that is trained to segment the road in the 2D topview representation of a lidar pointcloud. This approach, however, regards only the road boundaries. In a similar manner, He et al. [15] apply a dual-view convolutional neural network to the topview perspective of an RGB image and extract lane boundaries for multiple lanes. Alvarez et al. apply their approach to road segmentation [16], but use a much simple neural network. In the same way, multiple approaches have been presented at the KITTI benchmark like [17], but all of them regard only the ego lane, since KITTI

[1]FZI Research Center for Information Technology, Karlsruhe, Germany {ameyer,salscheider,orzechowski}@fzi.de
[2]Institute for Measurement and Control Systems, Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany stiller@kit.edu

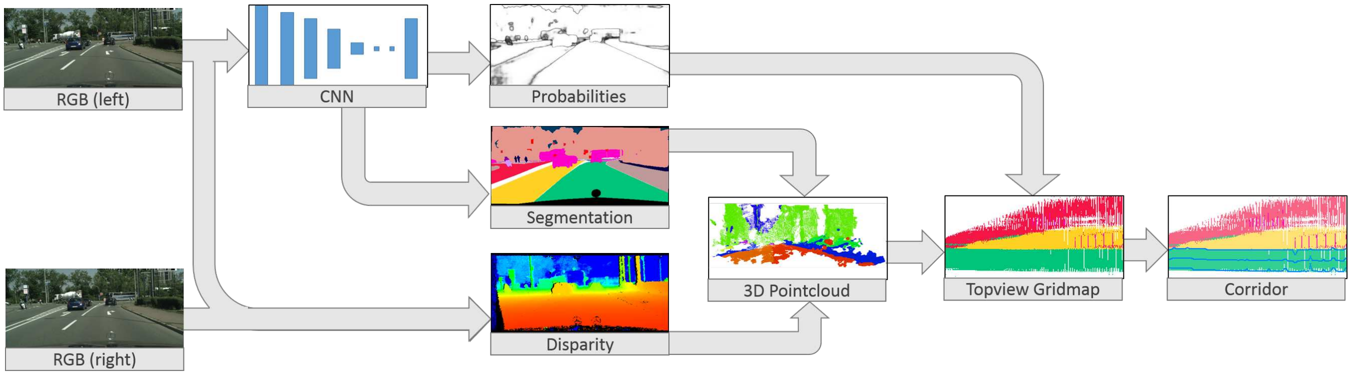

Fig. 2. Perception Pipeline for Mapless Driving on an Ego-Lane Corridor. Each left image of the stereo system is segmented by the neural network into the lanes of the road ahead. Using the right RGB image a semantic 3D pointcloud is produced that is transformed into a topview grid map. The resulting corridor is spanned across the grid map cells assigned with the ego label.

does not provide other labels.

Two more recent approaches, estimate the drivable path instead of the lane borders with regards to obstacles and traffic rules ([9], [18]). Barnes et al. [18] utilize a SegNet trained with weakly-supervised data generated by the driven path projected onto past images. The disadvantage of both works is that they do not generate a representation of the environment, but estimate a drivable path. This way a subsequent planning module can not regard the actual drivable area, but only the limited output. On the contrary, our scene representation allows to calculate a flexible path, based on the actual environment.

The result of the mentioned approaches is mainly presented in the front-view perspective of the vehicle. In this paper, we decided on using the birds-eye or top-view perspective in order to enable further processing for planning and driving. Other approaches dealing with top-views ([6], [14], [15], [19]) are not focusing on the semantics of the lane representation.

Among all the aforementioned approaches only a few ([3], [5], [7], [10], [15]) consider more than a single lane. In comparison, our approach estimates the semantic of the neighboring lanes (e.g. oncoming or neighboring lane in the same direction) and is able to work stable when markings and distinct curbs are not present. Additionally we can cope with construction sites and rail tracks.

To this end, we address the issue of mapless driving by proposing an efficient lane detection system using a deep neural network. Our system abolishes the dependency on highly accurate maps for roads (no intersections) and is able to represent the lanes purely based on RGB images. However, for routing on urban roads any coarse navigational input either from a less accurate map or user input is still necessary.

The approach features several stages. First, we obtain a pixel-wise semantic segmentation of the current scene using a deep neural network. The output of our network includes the lane geometry as well as the semantics. The data is aggregated into a top-view of the current scene allowing for a compact lane representation. An example scene with our results is illustrated in Fig. 1. Overall, our major contributions are the following:

1) Segmentation concept and dataset for lanes and
2) Pipeline for online lane segmentation as alternative to highly accurate maps.

B. Datasets

In the research area of lane segmentation, to our knowledge no research with focus on the semantics of a lane has been established. Therefore, datasets containing relevant lane information are rare. The KITTI-ROAD dataset [19] contains ego lane information. Although it has been extended by [20], less than 120 training images are available for training on geometric ego lanes, but the dataset lacks semantic information about other lanes.

We therefore extend the popular Cityscapes dataset [21]. This dataset contains image sequences showing street scenes in German cities and their semantic segmentation for a single image within a sequence. The segmentation covers 19 classes including a *road* class, that spans all areas drivable by a car.

II. Our Method

In order to push mapless driving, we present an approach inferring the lane semantics in a single step based on an RGB image. The resulting scene representation is used to extract lane boundaries for the ego lane including the semantic information about adjacent lanes. This information is crucial for deciding, whether it is feasible to change a lane based on the current traffic rules and lane topology also in unusual situations like construction sites. Additionally, we also provide other Cityscapes classes including obstacles on the lane, in order to further contribute to the overarching scene representation. An overview of our perception pipeline for lane segmentation is presented in Fig. 2.

A. Dataset for Lane Semantics

For our approach we manually extended parts of the Cityscapes dataset. In total we used 1974 training and 443 test images. The extension comprises four classes concerning the lane layout: *ego, parallel and opposite*. The ego lane

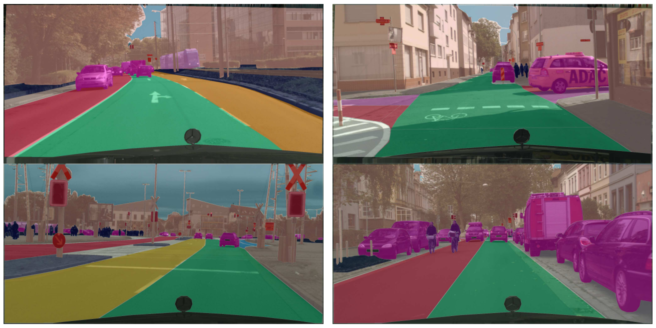

Fig. 3. An excerpt from the dataset visualizing our lane semantics extension. *Red*: The opposite lane. *Yellow*: The left lane parallel to ego. *Orange*: The right lane parallel to ego. A vehicle could change onto both parallel lanes. *Green*: The Ego Lane. *Purple*: Target roads at intersections, that are not evaluated in this work. *White*: Road parts that do not have semantics for vehicles. Remaining classes from the Cityscapes dataset are also illustrated. In our dataset, static background objects from Cityscapes are combined to a single class.

is defined as the lane, the vehicle is driving on. In doubt at intersections, we chose the lane going straight to be the ego lane. We define all lanes directing to the same direction like the ego lane as *parallel*. For on-coming traffic the *opposite* label is assigned. Lanes turning to the left or right at intersections are labeled with a *target* label, but are not evaluated here. All drivable road parts have been classified as one of the aforementioned classes. Road parts that are semantically not drivable remain with a general *road* label. In case of unmarked roads we assume the border between *ego* and *opposite* to be the centerline of the road. Some example images with the ground truth labels are shown in Fig. 3.

B. Semantic Lane Segmentation

As preparation for our approach we trained a neural network commonly used in semantic segmentation. We base our network on the ResNet-38 Architecture [22], that is able to segment street scenes semantically. Our network structure is given in Table I.

Lane semantics are not only encapsulated in the road parts, but also in e.g. traffic signs or the driving direction of cars. Therefore, we retain Cityscapes labels like *vehicle*, *pedestrian*, *traffic sign* and *traffic light* in our dataset. The remaining static labels, e.g. *pole* or *building*, are combined to a single *static* class, for simplicity

We use the trained network for predicting semantic labels for the left image of a stereo camera system. For each pixel in the image, the neural network estimates a pseudo-probability and a class. Both are used in the further processing.

C. Topview and Grid Map Projection

In parallel, our stereo camera system [23] computes a disparity image which is synchronized with the labeled output image of the neural network. Using the calibration parameters the disparity values are transformed into a single 3D point cloud, where each point gets assigned the corresponding label. By projecting the labeled point cloud onto the ground plane of the vehicle coordinate system, we obtain

	Convolution (7x7)	64 channels, stride 2
	Maxpooling (2x2)	
3x	ResNet module	64 channels
	Maxpooling (2x2)	
4x	ResNet module	128 channels
3x	ResNet module	256 channels, 2x dilation
1x	ResNet module	256 channels, 4x dilation
1x	ResNet module	256 channels, 8x dilation
1x	ResNet module	256 channels, 4x dilation
3x	ResNet module	512 channels
3x	Deconvolution (2x)	64 channels
	Convolution (1x1)	23 channels
	Softmax	

TABLE I

THE STRUCTURE OF OUR RESNET ARCHITECTURE. BATCH NORMALISATION AND RELU LAYERS ARE USED IN THE USUAL PLACES.

a topview grid map representation. Since we are interested in a road focused representation, all 3D points with a height greater than a threshold λ are discarded[1].

Depending on the resolution of the grid map[2], multiple pixels, especially in the foreground of the image, could be assigned to the same grid cell. in order to avoid this, each cell of the grid map stores the probabilities for each assigned label. Each grid cell $c_{x,y}$ in the grid map gets assigned the label $l_i \in L$ with the highest sum of the label probabilities $p_I(x, y)$ estimated by the network for an image I at the topview position (x, y). Hence, a cell's value is defined by

$$c_{x,y} = \arg\max_{l \in L} \sum_{I_{x,y}} p_I(x, y) \cdot 1(I_{x,y} = l), \qquad (1)$$

where $1(\cdot)$ denotes the indicator function. This way, the grid map stores only the most likely label for each cell, where multiple labels have been assigned.

D. Temporal Consistency

Temporal information of previous frames is important in order to maintain a continuous representation. This is especially important for estimating lane semantics in street scenes. Since evaluating multiple frames with a neural network or maintaining a state over time is time consuming, this work proposes to apply the temporal consistency in a post-processing step.

Based on the grid map insertion method above, multiple frames can be fused using the estimated label probabilities.

We utilize the odometry of the vehicle to transform the grid maps into a global coordinate system. This way, two labels from the same frame s in sequence S and values from adjacent frames s_i and s_j both assigned to cell at (x, y) are treated the same when processed by the grid map. Hence, extending (1) with a temporal component a cell's value

$$c_{x,y} = \arg\max_{l \in L} \sum_{I \in S} \sum_{I_{x,y}} p_I(x, y) \cdot 1(I_{x,y} = l) \qquad (2)$$

is summed over multiple labeled data from consecutive RGB frames.

[1] In this work, $\lambda = 0.5m$

[2] In this work, grid cells of size 10cm on an area of 100x100m

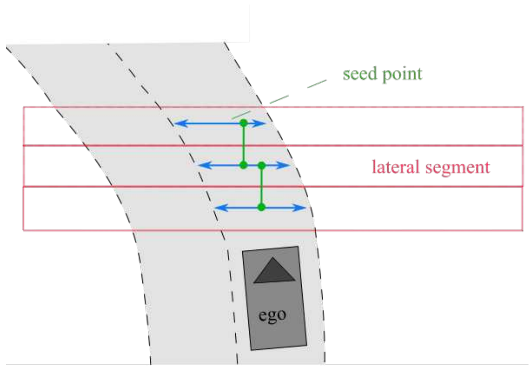

Fig. 4. The area ahead of the vehicle is divided into lateral segments and for each segment we determine the width of the area classified as ego lane. The seed point for the algorithm is determined for each segment based on the corridor from the previous segment. As initial seed point the center of the previous corridor is used. This way, all forward facing road layouts can be represented.

To avoid overwriting static labels from the current frame with dynamic objects from a last frame, we exclude all dynamic classes from the temporal storage and only use dynamic labels from the current frame.

E. Corridor Extraction

Given the topview grid map we further extract a corridor that spans the aspects of the road labeled as ego lane. Using the gridmap created with our spatio-temporal fusion, the resulting corridor is more robust and consistent over time. A corridor directly extracted from the frontview images would be highly prone to inconsistent corridors between different frames.

In order to extract the corridor, we divide the area ahead of the vehicle in lateral segments equally distributed with a fixed size. This step is illustrated in Fig. 4.

Similar to region growing, we move the border of the corridor to the left and right within each segment until the semantic label changes. Thus, only reachable areas are included in the resulting corridor. In order to enable not only straight corridors to be extracted, but also curved ones, the seed point for the algorithm is determined for each segment based on the corridor from the previous segment. As initial seed point the center of the previous corridor is used. Therefore, all forward facing road shapes can be represented by the resulting corridor independent of the unusual or unexpected shape, that may occur due to reconstructions. Compared to model fitting approaches, unusual road layouts and frequent changes in the curvature can be represented easily.

III. EVALUATION

We evaluate our approach on the Cityscapes images of Frankfurt, containing 266 images in total. We apply our approach to the full resolution images of 1024x2048 pixels. For the evaluation we regard both the front-view and top-view perspective, also called birds-eye-view (BEV). In contrast to a front-view comparison the BEV enables analyzing our fusion and it therefore allows to interpret the measured error in terms of continuous planning and driving [19].

TABLE II
INTERSECTION OVER UNION FOR LANE CLASSES

	Ego	Parallel	Opposite	IoU	IoU_{lane}
Front-View	80.01%	46.46%	48.21%	64%	58.26%
Multi BEV	80.02%	53.72%	58.66%	-	64.13%

We provide front-view, pixel-wise segmentation results using the intersection over union (IoU) metric, which is widely used in many vision benchmarks, e.g. Cityscapes [21]. Let TP denote the true-positive, FP the false-positive, and FN the false-negative values for each class $l \in L$. Then, the IoU metric for a specific label is

$$\overline{IoU}_l = \frac{TP_l}{TP_l + FP_l + FN_l} \tag{3}$$

and the weighted IoU metric for a set of classes L is

$$\overline{IoU}_{class} = \frac{\sum_l IoU_l}{|L|}. \tag{4}$$

This metric has the advantage that all classes are represented equally independent of their size in the dataset. Since the focus of this paper is on lane detection we further restrict $\overline{IoU}_{class}$ to the lane classes and refer hereby to as $\overline{IoU}_{lane}$. This allows us to evaluate only the lane segmentation.

For the evaluation, we utilize the validation data from the Cityscapes dataset, which, in addition to semantic label ground truth, contains disparity images, the corresponding right image and 10 images before and after the labeled one. Using this data, our pipeline can process both the ground truth and our segmentation results. We convert the ground truth images given by the Cityscapes dataset into the top-view perspective using the provided disparity values and camera calibration parameters. Having this representation of both the ground truth data and our results, we can easily apply the metrics to the BEV space.

A. Semantic Segmentation of Lanes

To compare our results with state-of-the-art segmentations we analyze the lane segmentation isolated from the overall pipeline. Tab. II summarizes the quantitative results we achieved on our dataset. On average we achieve an $\overline{IoU}_{lane}$ of 58.26% on all lane classes. For the ego lane, which is the main focus for driving, we achieve 80.01% on average. In spite of a different focus and different datasets, we compare our results to [18], because their approach is mostly related. They achieve an IoU up to 85% on the KITTI dataset. Here we highlight, that the KITTI dataset only contains simple roads and no intersections, whereas Cityscapes contains intersections and up to four lanes in one direction. An excerpt from the segmentation results is shown in Fig. 5 and in the attached video file showing a sequence from the demo video of Cityscapes.

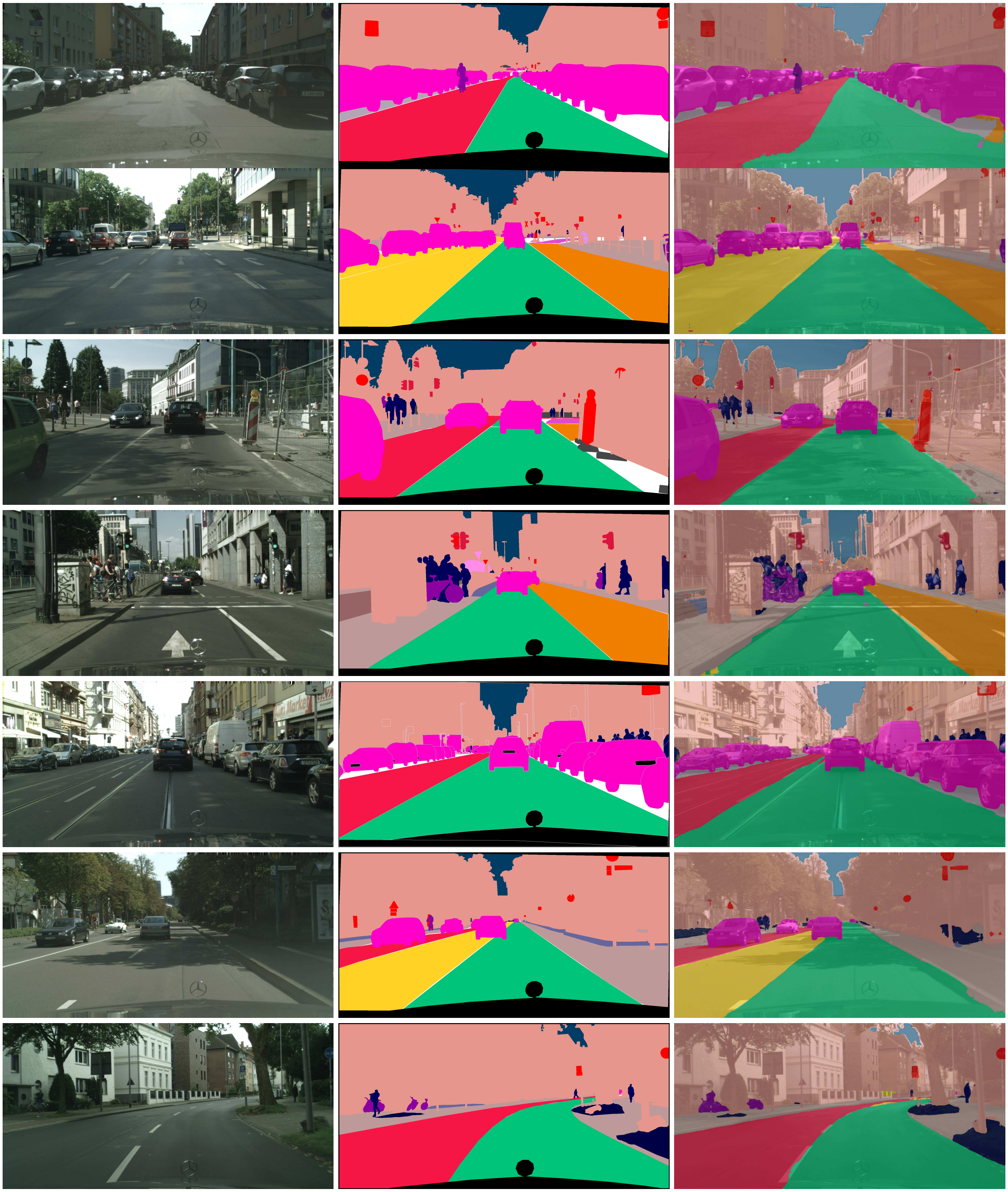

Fig. 5. Example images from the segmentation of semantic lanes. We show that we can handle both non-marked roads (third row), vanished markings (7th row) and construction sites (second row). The first column shows the scene image, the middle column visualizes the labels including the lanes and the right column shows the resulting segmentation from our approach. The colors are as follows. *Yellow*: The left lane parallel to ego. *Right*: The right lane parallel to ego. *Green*: Ego Lane. *Red*: Opposite Lane. *White*: Road parts that do not have semantics for vehicles. *Pink*: All road vehicles, e.g. cars. As can be seen, we are in fact able to identify multi-lane roads precisely even in the absence of lane markings and presence of tram rails.

B. Top-View Evaluation

As described in Sec. II our approach works best for sequences of images, because we utilize the temporal movement of the vehicle in order to enhance the grid map. A single frame approach leads to top-views with very sparse information towards greater distances as can be seen in Fig. 6 on the top. This problem is solved by our approach with the multi-frame accumulation of the grid map, where multiple frames captured from different view angles reduce the sparsity.

To apply our pipeline to the Cityscapes sequences we require a precise ego motion between two adjacent frames. Using the yaw rate, speed and timestamps of each image an accurate vehicle motion could be estimated. A resulting top-view built from about 10 frames per sequence is shown in Fig. 6 on the bottom. In real scenarios the topview is even more dense, when dealing with a continuous stream of images.

To evaluate our accumulation approach for the grid map, we calculate in addition to the IoU for all lanes classes also the precision, recall and F1-Score for ego lane. The results are shown in Tab. II and III. As can be seen from the results, we could improve the results of the single-shot front-view segmentation with our multi-frame topview approach for the separate lane classes.

Since we developed our own dataset it is hard to compare our results with others. However, [14], [24] and [17] use the KITTI benchmark in order to estimate road and ego lane borders, but without any further semantic information, resulting in an F1-score of up to 94.07% resp. 90.54% and 89.88%, representing the current evaluation top 10 at the KITTI ROAD Benchmark. Our results are therefore within the range of current KITTI submissions. However, when regarding our dataset, we could clearly state, that the images of Frankfurt available for validation are far more complex than the KITTI dataset with a single ego lane. Additionally we provide further details on the road semantics, that facilitate even more robust driving.

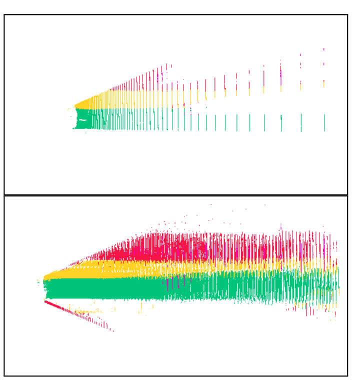

Fig. 6. Comparison between single-shot (top) and multi-frame (bottom) top-view images. The multi-frame has a higher density over the single-shot top-view since the data is aggregated over time. The input sequence is in Frankfurt and is shown in Fig. 5 in the last row.

TABLE III
BEV METRICS FOR COMPARING EGO LANE SEGMENTATION RESULTS

	F1	Precision	Recall
Lidar CNN [14]	94.07%	92.81%	95.37 %
RBNet [24]	90.54%	94.92%	86.56%
Up-Conv-Poly [17]	89.88%	92.01%	87.84%
Ours (Front-view)	88.89%	89.61%	88.19%
SPRAY [25]	83.42%	84.76%	82.12%

IV. CONCLUSIONS

In this work, we presented a deep neural network to extract semantic lane information in order to reduce the dependency on maps. We estimate not only the lane geometry but also the semantics of each lane (ego lane, ego-parallel lane and opposite lane) purely based on RGB images. A post-processing step refines the output and aggregates information in a top-view.

We show that we are able to robustly estimate the ego lane with an IoU of 80% on different street scenes. Our approach also copes with difficult road structures, including one way streets and streets with up to four lanes. Furthermore, our method can deal with incomplete or missing road markings and tram rails.

In future work, we will extend the work on other representations for the lane and also improve the support for intersections. Since we further work on the topic, we will also extend the dataset.

ACKNOWLEDGMENT

This work was funded by Daimler AG as part of the Tech Center a-drive initiative. The authors thank Daimler AG for the fruitful collaboration and the support for this work.

REFERENCES

[1] Ö. Ş. Taş, N. O. Salscheider, F. Poggenhans, S. Wirges, C. Bandera, M. R. Zofka, T. Strauss, J. M. Zöllner, and C. Stiller, "Making Bertha Cooperate-Team AnnieWAY's Entry to the 2016 Grand Cooperative Driving Challenge," *IEEE Transactions on Intelligent Transportation Systems*, vol. PP, no. 99, pp. 1–15, 2017.

[2] E. D. Dickmanns and B. D. Mysliwetz, "Recursive 3-D road and relative ego-state recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 199–213, 1992.

[3] C. Fernandez, R. Izquierdo, D. F. Llorca, and M. A. Sotelo, "Road curb and lanes detection for autonomous driving on urban scenarios," in *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1964–1969, 2014.

[4] M. Beyeler, F. Mirus, and A. Verl, "Vision-based robust road lane detection in urban environments," in *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4920–4925, 2014.

[5] 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.

[6] J. Li, X. Mei, D. Prokhorov, and D. Tao, "Deep Neural Network for Structural Prediction and Lane Detection in Traffic Scene," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 3, pp. 690–703, 2017.

[7] J. Beck and C. Stiller, "Non-parametric lane estimation in urban environments," in *IEEE Intelligent Vehicles Symposium (IV)*, pp. 43–48, 2014.

[8] J. M. Alvarez, T. Gevers, and A. M. Lopez, "3D Scene priors for road detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 57–64, 2010.

[9] C. Guo, J. Meguro, Y. Kojima, and T. Naito, "A Multimodal ADAS System for Unmarked Urban Scenarios Based on Road Context Understanding," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 4, pp. 1690–1704, 2015.

[10] S. Sivaraman and M. M. Trivedi, "Dynamic Probabilistic Drivability Maps for Lane Change and Merge Driver Assistance," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2063–2073, 2014.

[11] 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.

[12] J. M. Alvarez, A. M. Lopez, T. Gevers, and F. Lumbreras, "Combining Priors, Appearance, and Context for Road Detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 3, pp. 1168–1178, 2014.

[13] D. Neven, B. De Brabandere, S. Georgoulis, M. Proesmans, and L. Van Gool, "Towards End-to-End Lane Detection: An Instance Segmentation Approach," *arXiv*, 2018.

[14] L. Caltagirone, S. Scheidegger, L. Svensson, and M. Wahde, "Fast LIDAR-based road detection using fully convolutional neural networks," in *IEEE Intelligent Vehicles Symposium (IV)*, pp. 1019–1024, 2017.

[15] B. He, R. Ai, Y. Yan, and X. Lang, "Accurate and robust lane detection based on Dual-View Convolutional Neutral Network," in *IEEE Intelligent Vehicles Symposium (IV)*, pp. 1041–1046, 2016.

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

[17] G. L. Oliveira, W. Burgard, and T. Brox, "Efficient deep models for monocular road segmentation," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4885–4891, 2016.

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

[19] J. Fritsch, T. Kühnl, and A. Geiger, "A new performance measure and evaluation benchmark for road detection algorithms," in *IEEE International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1693–1700, 2013.

[20] G. Mattyus, S. Wang, S. Fidler, and R. Urtasun, "HD Maps: Fine-Grained Road Segmentation by Parsing Ground and Aerial Images," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3611–3619, 2016.

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

[22] Z. Wu, C. Shen, and A. van den Hengel, "Wider or Deeper: Revisiting the ResNet Model for Visual Recognition," *arXiv*, 2016.

[23] B. Ranft and T. Strau, "Modeling arbitrarily oriented slanted planes for efficient stereo vision based on block matching," in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 1941–1947, 2014.

[24] Z. Chen and Z. Chen, "RBNet: A Deep Neural Network for Unified Road and Road Boundary Detection," in *Neural Information Processing*, Lecture Notes in Computer Science, pp. 677–687, Springer, Cham, 2017.

[25] T. Kühnl, F. Kummert, and J. Fritsch, "Spatial ray features for real-time ego-lane extraction," in *Intelligent Transportation Systems (ITSC), 2012 15th International IEEE Conference On*, pp. 288–293, IEEE, 2012.