Bird's Eye View Segmentation Using Lifted 2D Semantic Features

Isht Dwivedi[1]
idwivedi@honda-ri.com
Srikanth Malla[1]
smalla@honda-ri.com
Yi-Ting Chen[†][1][2]
ychen@nycu.edu.tw
Behzad Dariush[1]
bdariush@honda-ri.com

[1] Honda Research Institute USA
[2] National Yang Ming Chiao Tung University

Abstract

We consider the problem of Bird's Eye View (BEV) segmentation with perspective monocular camera view as input. An effective solution to this problem is important in many autonomous navigation tasks such as behavior prediction and planning, being that the BEV segmented image provides an explainable intermediate representation that captures both the geometry and layout of the surrounding scene. Our approach to this problem involves a novel BEV feature transformation layer that effectively exploits depth maps to transform 2D image features to the BEV space. The framework includes the design of a neural network architecture to produce BEV segmentation maps using the proposed transformation layer. Of particular interest is evaluation of the proposed method in complex scenarios involving highly unstructured scenes that are not represented in static maps. In the absence of an appropriate dataset for this task, we introduce the EPOSH road-scene dataset that consists of 560 video-clips of highly unstructured construction scenes, annotated with unique labels in both perspective and BEV. For evaluation, we compare our approach with several competitive baselines and recently published works and show improvement over state of the art on the Nuscenes and on our EPOSH dataset. We plan to release the dataset, code and trained models at https://usa.honda-ri.com/eposh.

1 Introduction

For intelligent mobility systems such as automated vehicles, driver assistance systems and mobile robots, Bird's-Eye View (BEV) segmentation can be used as a compact representation of the surroundings scene to support various decision making processes, including prediction, path planning, collision avoidance and navigation. The segmented BEV image provides an intermediate representation in the form of a semantically meaningful image that captures the spatial relationships of static and dynamic elements in an overhead view.

[†] Work done during Yi-Ting Chen's employment at Honda Research Institute USA

The BEV representation is generally preferred in many autonomous navigation tasks since the top-down view captures the essential spatial relationships on the ground plane. Furthermore, BEV semantic segmentation is an alternate, and often more useful method to monocular SLAM for online mapping, particularly in places where no map is available or where the map is updated due to unstructured events such as construction, traffic incidents, and unexpected debris on the path. On road scenes, a standard practice is to rasterize static elements in HD maps into a BEV image and combine with scene elements which change over time. An important characteristic of the BEV image is that it provides an intermediate representation that is interpretable and therefore facilitates identification of inherent failure modes in various upstream and downstream tasks. In addition, such a representation can easily consume perception results from perspective views and is extensible to other modalities which simplifies late fusion tasks.

While data generated from active sensors such as LiDAR are inherently metric in 3D and lend themselves well for BEV representation, their limitations include sparse measurements, low scan rates, and prohibitively high cost. Surround view cameras have a ubiquitous presence due their low-cost and are more suitable for generating semantically meaningful and dense per-pixel representation of the surrounding scene. Although stereo camera systems can be used to generate a three dimensional representation of the scene, their calibration and synchronization process is non-trivial and difficult to scale. To overcome the aforementioned challenges with active and stereoscopic sensing, this paper focuses on monocular BEV segmentation from egocentric views captured from a moving platform.

As compared to perspective semantic segmentation, monocular BEV segmentation is significantly more challenging for several reasons. First, while the number of pixels in the perspective images falls quadratically with radial distance from the camera, when transformed to the BEV space, farther regions occupy the same footprint as nearby regions. This makes prediction of farther away points increasingly difficult. A second challenge is that monocular depth estimation, an intermediate step in BEV segmentation, is an ill-poised problem because there can be multiple plausible depth estimates for a given single image. This issue exists for all monocular 3D scene understanding methods, including those that do not use depth maps explicitly. Finally, dealing with partial and self occlusions of objects in the scene (e.g. self occlusions from several vehicles) is a difficult and un-resolved problem. Effective BEV segmentation methods must reason about the shape of the complete object in the BEV, given the perspective view.

Existing works such as PON [23], VED [14], and VPN [19] extract 1D intermediate features, but this process does not fully capture essential spatial information present in the input image. [24] project each voxel volume onto the image to extract features. In their approach, similar features would be present along every ray emitting from the camera, causing depth ambiguity. Our proposed transformation layer is most similar to [24] but our approach overcomes the aforementioned limitations.

The main contributions of this work are as follows. First, we propose a novel transformation layer that effectively exploits depth maps to transform 2D image features to the BEV space. We also design a multi-scale neural network architecture to produce BEV segmentation maps using this transformation layer. We use perspective semantic segmentation as an auxiliary task which improves BEV segmentation performance. We are interested in applying this method for personalized navigation using distributed fleet of networked vehicles that communicate semantically meaningful and explainable information about unstructured and events that are not available in static maps. In the absence of an appropriate dataset for this task, we introduce the EPOSH road-scene dataset that consists of 560 video-clips of highly unstructured construction scenes, annotated with unique labels in both perspective and BEV. The dataset also contains instance based annotations for relevant classes such as lanes, construction cones, construction signs, lane lines etc. Finally, we evaluate our method on 2 large scale datasets that demonstrate efficacy of the proposed approach by improving upon the current state of the art performance.

2 Related Work

2.1 BEV Semantic Segmentation

Advances in perspective-view semantic segmentation algorithms [4, 31] are largely attributed to high-quality and large-scale semantic segmentation datasets such as Cityscapes [6] and Mapillary Vistas [17]. Recently, BEV segmentation has attracted attention because it offers a compact representation that captures the spatial configuration of road scenes [14, 18, 19, 21, 23]. Many of these approaches, including ours, first extract an intermediate representation from the input image using a CNN and then apply a transformation to convert these features to a 3D feature map. This approach has also been used for 3D bounding box estimation. The work in [24] is most similar to our work, but their method does not use depth information during transformation, which leads to large ambiguity in depth during prediction. Another approach [29, 30] is to convert the image to a Pseudo-LiDAR and then apply LiDAR sensor based approaches.

2.2 Semantic Mapping

Ongoing works on semantic mapping for outdoor navigation provide semantic abstractions of traffic scenes [16, 20, 27, 28]. Current methods [16, 20, 28] typically use multi-modal sensory inputs (i.e., LiDAR and camera) to obtain geometric and semantic information for semantic mapping. In [27], a camera-based semantic mapping framework with a flat world assumption is proposed. In this work (and recent camera-based systems), we do not assume a flat world constraint toward practical applications. To obtain the BEV representation of a local map given a perspective-view image, we propose estimating depth [10] explicitly and reconstructing the corresponding 3D road scenes using the estimated depth. Furthermore, we propose a novel framework that jointly considers perspective and BEV segmentation.

Topology related		Planning related	
Class name	Attributes	Class name	Attributes
⋆ Lane	Ego lane / not ego lane	⋆ Construction	cone sign barrier
⋆ Lane Line	Color, solid / dashed, single / double		
⋆ Crosswalk	Orientation	⋆ Blocked area	-
⋆ Inside Intersection area	3-way / 4-way	⋆ PErmissible Drivable Area (PEDA)	Branch types {Ego,1,2,3}
⋆ Road curb line	-		
⋆ Symbolic Road Marking	Straight, Left, Right, Straight-Left, Straight-Right	⋆ Lane affordance	Straight, left turn, right turn left lane, right lane change

Table 1: Classes and corresponding attributes in the EPOSH road-scene dataset

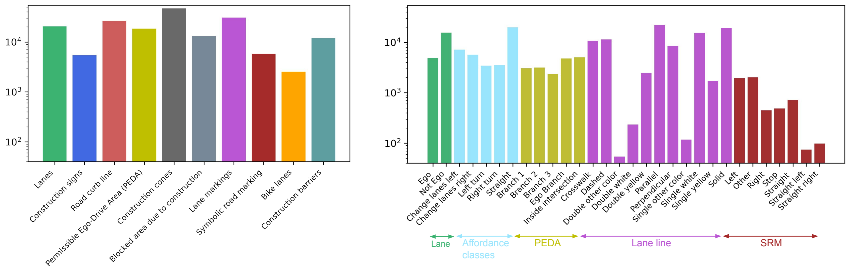

Figure 1: Figure showing distribution of classes and corresponding attributes in the perspective EPOSH dataset. The left subplot shows classes and the right subplot shows the corresponding attributes and affordance classes in the dataset.

2.3 Pseudo-LiDAR

Since the advent of LiDAR sensors in the autonomous car industry, considerable research effort [12, 13, 22] has been spent to build 3D scene understanding models like 3D bounding box estimation to efficiently utilize LiDAR data. To be able to use these advances with RGB image data, [29] first introduced Pseduo-LiDAR for 3D bounding Box estimation using a stereo camera setup. [30] has used a monocular camera setup to produce Pseudo LiDAR for 3D bounding box estimation by using a monocular depth estimation neural network. In this work, we also used a monocular camera setup to produce a Pseudo-LiDAR point cloud. A pixel-wise depth map generated using a CNN is used to project each pixel in the image to 3D world coordinates using camera intrinsics. Supervised depth estimation methods such as [15] aim to regress per pixel depth during training while self-supervised methods [1, 9, 10] use a geometry consistency loss.

2.4 Road Scene Datasets for BEV Segmentation

Existing road scene BEV segmentation algorithms built on BEV ground truth segmentation generated from datasets with High Definition (HD) maps and annotations of 3D bounding boxes for vehicles, pedestrians, and other traffic participants. The NuScenes [2] and Argoverse dataset [3] are the key enablers for the developments. The two datasets mainly focus on statics objects (e.g., lane marking, zebra crossing, and traffic light), and dynamic elements (e.g., vehicles and pedestrians). An important yet under-explored element is unstructured events such as construction areas, traffic incidents, and unexpected debris because they are indispensable for behavior prediction and planning modules. To this end, we propose the EPOSH road-scene dataset with road-topology and motion planning related class labels (details can found in Sec. 3) for BEV segmentation around construction scenes.

3 EPOSH Road-Scene Dataset

The dataset consists of 560 ego-centric road-scene video clips collected in the San Francisco Bay Area with a front-facing GoPro Hero 7 camera mounted on a vehicle. Most video clips are between 10 – 30 sec long and are recorded around construction zones, an unstructured scene that is under-explored in the existing literature. To understand the semantic context

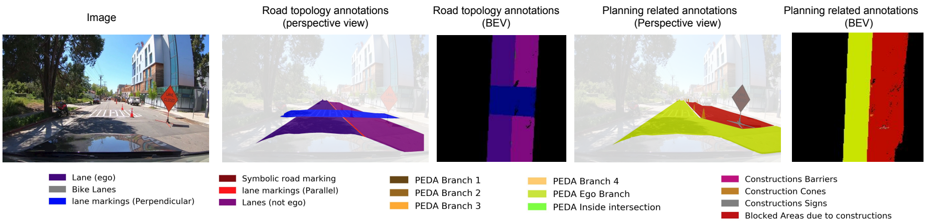

Figure 2: Figure showing a sample annotation from the EPOSH road-scene dataset

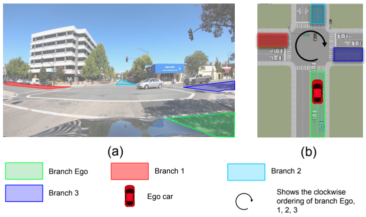

Figure 3: Figure showing an illustration of PEDA attributes for (a) perspective view and (b) BEV for the EPOSH road-scene dataset.

	EPOSH	NuScenes
BEV annotations	✓	✓
Perspective view annotations	✓	✗
Detailed construction zone attributes	✓	✗
Detailed road topology attributes	✓	✗
Intersection branch annotation	✓	✗
Affordance annotation	✓	✗

Table 2: Comparison of EPOSH with NuScenes

around construction zones, we propose unique labels that are categorized into two groups: road topology and planning as depicted in Figure 2. Both the perspective and BEV perspective images are labeled with the same classes. Table 1 summarizes all labels and their corresponding attributes. The supplementary material has additional details.

Given a video clip, about ten frames are selected manually and annotated with polygons. Each polygon is labeled with a class and the corresponding attributes. A total of 5,630 perspective images are annotated. To obtain the corresponding BEV annotations, we first apply COLMAP [25, 26] to reconstruct a 3D dense point cloud given a video clip. We observed that COLMAP is robust to the presence of moving elements in our experiments. We then annotate semantic labels of 3D points manually. Irrelevant classes such as trees, sky and buildings are removed. Given the estimated camera pose for each input frame, the BEV plane is calculated, and the annotated point cloud is projected to the ground to form BEV semantic segmentation. A total of 70,000 BEV perspective frames are constructed.

Table 2 compares the EPOSH dataset with NuScenes dataset for the task of understanding road scene in the perspective view and BEV. Figure 3 explains the attributes used for the class PErmissible Drivable Area (PEDA). Figure 1 shows a sample annotation from the EPOSH dataset. Please view supplementary material for more samples of the dataset.

4 Methodology

Given an input perspective image I, we predict the corresponding BEV segmentation map S_{bev} and perspective semantic segmentation map S_{per}. Figure 4 shows the proposed network architecture. Multi-scale features F are extracted from image I using a ResNet50 [11]. The features F are used both by the BEV feature transform layer and the Auxiliary-CNN for perspective segmentation as depicted in Figure 4. The BEV transform layer transforms features F in perspective space to Q in 3D space. The features Q are utilized to produce BEV segmentation. In the following sections, we describe our framework in detail.

4.1 Voxelized Pseudo-LiDAR Point Cloud Generation

It is challenging to directly predict BEV segmentation from a perspective image. A novel BEV transformation layer is proposed to tackle the challenging task. Specifically, given an input perspective image I, we estimate its depth map D using recent monocular depth estimation algorithms [10, 15]. At a pixel coordinate (u,v) in I, we unproject the coordinate to a 3D coordinate $P_{3d} = (X,Y,Z)$ using the camera intrinsics and the predicted depth map D, $X = (u - c_x)Z_{uv}/f_x$; $Y = (v - c_y)Z_{uv}/f_y$; $Z = D[u,v]$. The Pseudo-LiDAR point cloud is voxelized to a voxel grid V with dimensions $H \times W \times D$.

4.2 BEV Feature Transform Layer

Every voxel V_j in Voxel grid V contains varying n number of 3D points $(X_i^j, Y_i^j, Z_i^j) \in V_j$ for $i \in [1, \cdots, n]$ from the Pseudo-LiDAR point cloud corresponding to the pixels (u_i^j, v_i^j) of the image I. We store the corresponding 2D pixel locations in Voxel grid $\tilde{V}$ resulting $(u_i^j, v_i^j) \in \tilde{V}_j$. Using bilinear interpolation on features F we can extract a feature vector f_i^j of length l_f, for each pixel (u_i^j, v_i^j) in a given voxel $\tilde{V}_j$. The feature corresponding to a voxel $\tilde{V}_j$ is the mean of all the feature vectors of length l_f, $Q_j = \frac{\sum_{i=1}^{n} f_i^j}{n}$. Thus we have a l_f length feature vector for each voxel in our 3d voxel grid. The entire feature map can be represented as a $4d$ feature map with size (h,w,d,l_f). Figure 5 shows a schematic of the BEV feature transform layer. We use the BEV-segmentation CNN on this feature map to generate the final BEV segmentation predictions. The first layer in BEV segmentation CNN is a linear layer that converts features from shape (h,w,d,l_f) to (w,d,l_f). This is followed by a stack of 8 residual blocks, a convolutional & bilinear upsampling layer respectively.

4.3 Loss

We use binary cross-entropy loss during training. Since the different classes in the NuScenes dataset can be overlapping, we pair each class with its own background class and produce

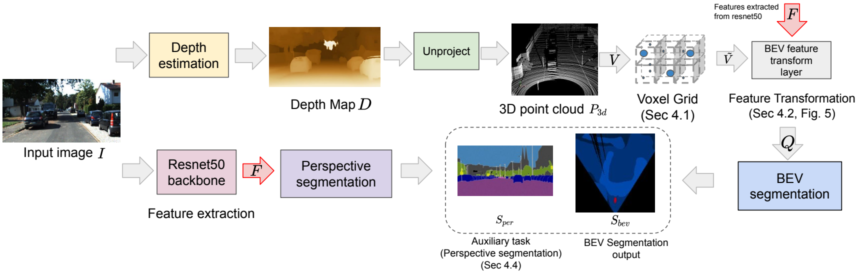

Figure 4: Network Architecture of our method. Features F (shown by red arrow) extracted using ResNet50 backbone are input to both the perspective segmentation block and BEV feature transform layer.

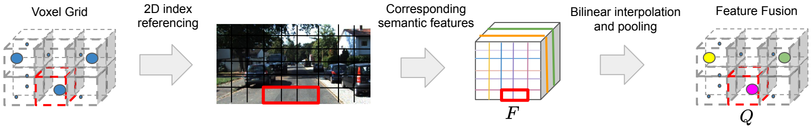
Voxel Grid
2D index referencing
Corresponding semantic features
F
Bilinear interpolation and pooling
Feature Fusion
Q

Figure 5: BEV feature transform layer. Features F and voxelized point cloud are input to this layer, and the output are transformed features Q.

2 maps for each class - a separate cross-entropy loss is calculated for each class. We also weight the loss of each class by the square root of the inverse class frequency. The weighted mean of all these losses is the total loss in these cases.

For the EPOSH road-scene dataset, the number of classes can be divided into 2 sets of non-overlapping classes as seen in the 2 columns of Table 1. We calculate the cross-entropy loss for each set of classes and then add them to get the final loss. Like the NuScenes dataset, we also weight each class by the square root of the inverse class frequency.

4.4 Auxiliary Task

During training, we added a small Auxiliary CNN which takes as input the resnet50 features F and predicts perspective semantic segmentation. The Auxiliary CNN is similar to the BEV segmentation CNN, consisting of a stack of 8 residual blocks followed by a convolutional layer. The loss used for back-propagation is, $L_{total} = L_{bev} + L_{per}$ where L_{per} and L_{bev} are the perspective semantic segmentation and BEV segmentation losses respectively.

5 Experiments

5.1 Datasets

We evaluate our method on 2 large scale datasets: NuScenes and EPOSH road-scene datasets. The NuScenes dataset has 1000 20 second video clips collected in Boston and Singapore. The dataset is recorded from 6 surround cameras along with a synchronized LiDAR sensor. It includes HD map annotations with classes such as drivable area, sidewalks, lanes etc. The dataset also has 3D bounding box annotations for cars and pedestrians. [23] first created a BEV segmentation dataset by incorporating the car and bounding box annotations on HD maps. We use this dataset for experiments on the NuScenes dataset. Refer to Section 3 for details about the EPOSH dataset.

5.2 Baselines and Evaluation

We use the mean Intersection over Union (IoU) over all classes for evaluation.

Published methods: We compare our method with recently published works on BEV segmentation - PON [23], VPN [19], VED [14]. We also compare with OFT [24] since it is similar to our method and also uses a transformation layer to convert monocular perspective image features to BEV features. To make these methods comparable with our method, only the feature transformation layer that converts features from perspective view to BEV is modified while keeping all other parts same.

Dataset / Pool Type	Dot	Avg	Max
NuScenes	20.2	**21.0**	20.5
EPOSH	26.8	**27.3**	27.1

Table 3: Effect of using different pooling methods in the BEV feature transform layer.

PointPillars (PP): This method is inspired from [13] which is used for 3d bounding box detection using LiDAR point clouds. Here, we adopt it here for our purpose. We use the Pillar Feature Net to extract a Pseudo Image directly from pointcloud P_{3d}. In addition to the 9 features used in [13], we also use 3 additional features for each point in the cloud - (r, g, b), i.e the color values corresponding to the each point (pixel). We use $0.625m^2$ sized bins. Finally, a CNN containing a stack of 8 residual blocks makes the final prediction.

Inverse Perspective mapping (IPM): This baseline is similar to [8]. We first use Deeplab v3+ [5] to predict semantic segmentation on the input image I and then use a homography to project the predicted map to the ground plane. For NuScenes dataset, we use a Deeplab v3+ model pretrained on cityscapes since it shares many classes with NuScenes BEV dataset. For the EPOSH dataset, we train a Deeplab v3+ model on the dataset directly since the dataset also contains perspective semantic segmentation annotations.

Depth Unprojection: For this baseline, we first compute perspective semantic segmentation on the input image I using the same models as used for IPM. We then project image points to 3D space using a depth map and then project each point to the ground plane. The semantic segmentation of each image point is thus transferred to the ground plane. For the NuScenes dataset, we use LiDAR ground truth as the depth map. For the EPOSH dataset, we use the same pretrained CNN [10] used by our model.

Ablations study: We explore different types of pooling layers used in the BEV feature transform later. Table 3 shows the results of our experiments. For dot product pool, we use a convolutional layer on feature F to predict a 2D attention map. Features are weighted by this map and then average pooled. For both the NuScenes and EPOSH datasets, we find that Average (Avg) Pool gives best results followed by Max pool and Dot product pool respectively. Thus, we use Avg pool for all our experiments.

5.3 Implementation details

For depth estimation on the NuScenes dataset, we trained a CNN [15] with ResNet18 backbone using LiDAR ground truth. For the EPOSH road-scene dataset, we used a pretrained model [10] that is trained in a self-supervised fashion. Thus, our model can be trained in the absence of LiDAR ground truth. All of our experiments are implemented using PyTorch and we use the PyTorch Scatter library[1] to implement the BEV feature transform layer. Following [23], we used a BEV grid extending 25 m to either side of the camera and 49 meters in the front. For both datasets, the BEV footprint of 49×50 meters is represented by a ground truth of shape 196×200 pixels. For voxelizing the point cloud, we used voxel grid of length 50 cm. All models in our work are trained for 40 epochs with initial learning rate and weight decay rate set to 10^{-3} and 10^{-4} respectively. We decay the learning rate 10 times at epochs 25 and 35. For the NuScenes dataset, since perspective segmentation ground truth is not available, we use pseudo-labels generated using a deeplab v3 model trained on Cityscapes [7].

[1] https://github.com/rusty1s/pytorch_scatter

	Drivable*	Ped. crossing	Walkway*	Carpark	Car*	Truck	Bus*	Trailer	Constr. veh	Pedestrian*	Motorcycle*	Bicycle*	Traf. Cone	Barrier	Mean	CS Mean
State of the art																
IPM	40.1	-	14.0	-	4.9	-	3.0	-	-	0.6	0.8	0.2	-	-	-	9.1
Depth Unproj.	27.1	-	14.1	-	11.3	-	6.7	-	-	2.2	2.8	1.3	-	-	-	9.4
VED [14]	54.7	12.0	20.7	13.5	8.8	0.2	0.0	7.4	0.0	0.0	0.0	0.0	0.0	4.0	8.7	12.0
PointPillars [13]	58.1	26.0	27.4	15.7	23.9	14.6	17.6	13.2	3.0	2.3	3.8	3.4	3.6	7.1	15.8	19.3
VPN [19]	58.0	27.3	29.4	12.9	25.5	17.3	20.0	16.6	4.9	7.1	5.6	4.4	4.6	10.8	17.5	21.4
PON [23]	60.4	28.0	31.0	18.4	24.7	16.8	20.8	16.6	**12.3**	**8.2**	7.0	**9.4**	5.7	8.1	19.1	23.1
OFT [22]	**62.4**	30.9	34.5	23.5	34.7	17.4	23.2	**18.2**	3.7	1.2	6.6	4.6	1.1	12.9	19.6	23.9
Ours																
Ours	62.3	31.8	37.3	25.2	37.4	18.7	24.8	16.4	4.7	3.4	7.9	7.2	3.9	**13.6**	21.0	25.8
With Aux Task	61.1	**33.5**	**37.8**	**25.4**	**37.8**	**20.4**	**31.8**	14.2	2.7	5.9	**10.5**	6.69	**7.57**	13.4	**22.1**	**27.4**

Table 4: BEV segmentation performance (IoU) comparison on the NuScenes dataset. * denotes the classes common between Cityscapes dataset and NuScenes dataset. CS Mean is the mean of only these classes.

	Topology related									Motion planning related						
	Ego lane	Non-ego lane	Bike lane	SRM	Crosswalk	Inside 3-way	Inside 4-way	Lane line	Topo B/G	Blocked area	PEDA	Const. cones	Const. signs	barriers	M.P.B/G	Mean
State of the art																
IPM	16.1	9.8	0.8	1.7	7.3	3.08	13.9	3.58	27.7	9.1	17.5	0.31	0.59	4.02	29.1	9.73
VED [14]	21.3	12.9	1.19	2.38	9.61	4.05	18.4	4.72	36.6	12.	23.1	0.41	0.77	5.29	39.9	12.8
Depth Unproj.	17.0	10.7	0.93	1.87	7.52	3.17	13.7	6.7	53.6	13.4	23.3	0.26	1.61	5.31	52.2	14.1
PointPillars [13]	34.7	19.1	7.3	3.2	10.9	0.9	27.3	8.1	64.3	19.4	37.4	0.20	1.4	5.4	69.4	20.6
VPN [19]	37.9	22.9	2.13	4.26	17.1	7.22	32.8	8.4	65.2	21.4	41.1	0.74	1.38	9.44	71.2	22.8
PON [23]	41.0	24.8	2.30	4.6	18.5	7.8	35.4	9.1	70.4	23.1	44.4	0.8	1.5	10.2	76.9	24.7
OFT [22]	45.4	25.9	12.0	5.2	14.6	1.48	32.4	10.5	69.9	26.3	54.9	0.42	2.7	5.61	81.2	25.9
Ours																
Baseline	45.9	**27.4**	**13.9**	**6.5**	**18.6**	1.11	**34.2**	10.9	70.2	29.7	**58.2**	**2.4**	2.17	**10.7**	82.1	27.3
With Aux Task	**46.8**	25.8	13.1	5.97	17.9	**2.47**	31.5	**12.5**	**70.7**	**30.1**	56.2	0.24	**2.58**	7.94	**82.5**	**27.8**

Table 5: BEV segmentation performance (IoU) comparison on the EPOSH dataset.

6 Results

Quantitative results: We compare our method with baselines and competing works on the NuScenes and EPOSH dataset in Table 4 and Table 5, respectively. IPM has the worst performance for both datasets. Depth Unproj. gives relatively better performance for EPOSH dataset compared to NuScenes possibly because LiDAR ground truth used for NuScenes is sparse but Pseudo-LiDAR used for EPOSH is more dense. VED, VPN and PON use an intermediate 1D feature representation. VED is not able to preserve any spatial information in intermediate features since it uses an variational Auto-Encoder type architecture and thus performs the worst out of these 3 methods. We posit that because PON retains the horizontal spatial image axis in the 1D feature, it performs the best among these methods. We note that PointPillars performs slightly worse compared to VPN, we believe that the Pseudo Image used in this method does not effectively represent the (r,g,b) color information of the input image. Since OFT preserves spatial information in its transformation, it is able to get better performance compared to PON. We notice that our Baseline is able to further improve upon OFT for both the datasets. Using perspective semantic segmentation as an Auxiliary task further improves performance for both datasets as seen in the last 2 rows of Table 4 and 5.

Qualitative results: Figure 6(a) highlights qualitative improvements achieved by our model due to the use of the Auxiliary task on the NuScenes dataset. Figures 7 and 6(b) compare our model with the most competitive published methods from Table 4 and 5 on both datasets.

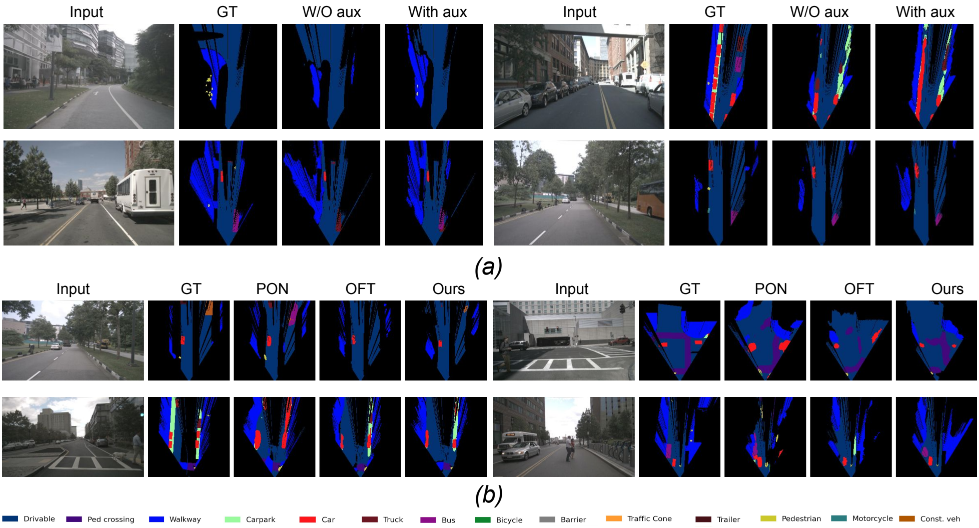

Figure 6: (a) Examples from NuScenes dataset comparing performance of our method with and without using the Auxiliary task, (b) Quantitative results on the NuScenes dataset. Circles emphasize the areas where there are major differences between different methods.

7 Conclusion

We considered the problem of BEV segmentation using our novel transformation layer that preserves spatial information and effectively exploits depth information. We demonstrated the efficacy of the proposed models with experimental evaluations on the NuScenes and EPOSH datasets. The results showed that our novel transformation layer improves on current state of the art methods through a more effective mapping of perspective image features to BEV features. In the future, we plan to add temporal modelling to our model and also plan to make the entire model including the depth estimation model end-to-end trainable.

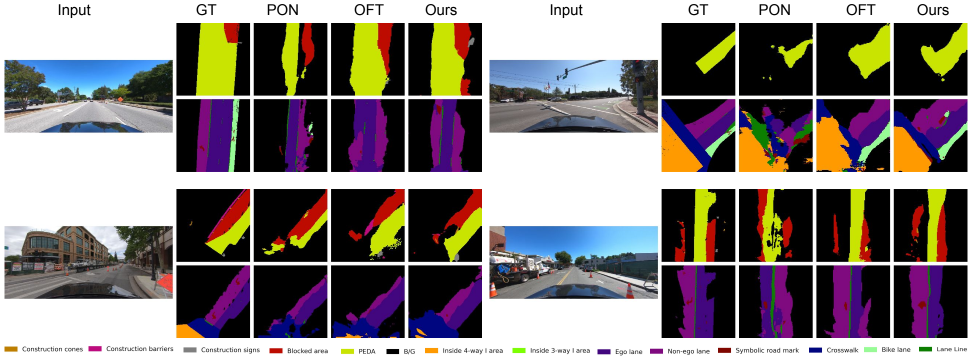

Figure 7: Quantitative results on the EPOSH dataset. Motion planning related and topology related classes are shown on the first and second row, respectively.

References

[1] Jia-Wang Bian, Zhichao Li, Naiyan Wang, Huangying Zhan, Chunhua Shen, Ming-Ming Cheng, and Ian Reid. Unsupervised scale-consistent depth learning from video. *International Journal of Computer Vision (IJCV)*, 2021.

[2] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuScenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020.

[3] Ming-Fang Chang, John Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Carr, Simon Lucey, Deva Ramanan, and James Hays. Argoverse: 3d tracking and forecasting with rich maps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8748–8757, 2019.

[4] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[5] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[6] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[7] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016.

[8] Liuyuan Deng, Ming Yang, Hao Li, Tianyi Li, Bing Hu, and Chunxiang Wang. Restricted deformable convolution-based road scene semantic segmentation using surround view cameras. *IEEE Transactions on Intelligent Transportation Systems*, 21 (10):4350–4362, 2019.

[9] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 270–279, 2017.

[10] Vitor Guizilini, Rares Ambrus, Sudeep Pillai, Allan Raventos, and Adrien Gaidon. 3d packing for self-supervised monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2485–2494, 2020.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pages 630–645. Springer, 2016.

[12] Jason Ku, Melissa Mozifian, Jungwook Lee, Ali Harakeh, and Steven L Waslander. Joint 3d proposal generation and object detection from view aggregation. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–8. IEEE, 2018.

[13] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12697–12705, 2019.

[14] Chenyang Lu, Marinus Jacobus Gerardus van de Molengraft, and Gijs Dubbelman. Monocular semantic occupancy grid mapping with convolutional variational encoder–decoder networks. *IEEE Robotics and Automation Letters*, 4(2):445–452, 2019.

[15] Fangchang Ma and Sertac Karaman. Sparse-to-dense: Depth prediction from sparse depth samples and a single image. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4796–4803. IEEE, 2018.

[16] Daniel Maturana, Po-Wei Chou, Masashi Uenoyama, and Sebastian Scherer. Real-time semantic mapping for autonomous off-road navigation. In *Field and Service Robotics*, 2018.

[17] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017.

[18] Mong H Ng, Kaahan Radia, Jianfei Chen, Dequan Wang, Ionel Gog, and Joseph E Gonzalez. Bev-seg: Bird's eye view semantic segmentation using geometry and semantic point cloud. *arXiv preprint arXiv:2006.11436*, 2020.

[19] Bowen Pan, Jiankai Sun, Ho Yin Tiga Leung, Alex Andonian, and Bolei Zhou. Cross-view semantic segmentation for sensing surroundings. *IEEE Robotics and Automation Letters*, 5(3):4867–4873, 2020.

[20] David Paz, Hengyuan Zhang, Qinru Li, Hao Xiang, and Henrik Christensen. Probabilistic semantic mapping for urban autonomous driving applications. In *Proceedings of International Conference on Intelligent Robots and Systems*, 2020.

[21] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *European Conference on Computer Vision*, pages 194–210. Springer, 2020.

[22] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 918–927, 2018.

[23] Thomas Roddick and Roberto Cipolla. Predicting semantic map representations from images using pyramid occupancy networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11138–11147, 2020.

[24] Thomas Roddick, Alex Kendall, and Roberto Cipolla. Orthographic feature transform for monocular 3d object detection. *In Proceedings of British Machine Vision Conference*, 2019.

[25] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016.

[26] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision*, pages 501–518. Springer, 2016.

[27] Sunando Sengupta, Paul Sturgess, L'ubor Ladický, and Philip H. S. Torr. Automatic dense visual semantic mapping from street-level imagery. In *Proceedings of International Conference on Intelligent Robots and Systems*, 2012.

[28] Chien-Yi Wang, Athma Narayanan, Abhishek Patil, Wei Zhan, and Yi-Ting Chen. A 3d dynamic scene analysis framework for development of intelligent transportation systems. In *Proceedings of IEEE Intelligent Vehicles Symposium*, 2018.

[29] Yan Wang, Wei-Lun Chao, Divyansh Garg, Bharath Hariharan, Mark Campbell, and Kilian Q Weinberger. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8445–8453, 2019.

[30] Xinshuo Weng and Kris Kitani. Monocular 3d object detection with pseudo-lidar point cloud. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019.

[31] Yuhui Yuan, Xilin Chen, and Jingdong Wang. Object-contextual representations for semantic segmentation. In *Proceedings of the European Conference on Computer Vision*, 2020.