Supplementary Material: Understanding Road Layout from Videos as a Whole

Buyu Liu[1] Bingbing Zhuang[1] Samuel Schulter[1] Pan Ji[1] Manmohan Chandraker[1,2]
[1]NEC Laboratories America [2]UC San Diego

This supplemental material contains the following details that we could not include in the main paper due to space restrictions:

- **(Sec. 1)** Details on feature aggregation
- **(Sec. 2)** Implementation details
- **(Sec. 3)** The IoU metric for evaluation
- **(Sec. 4)** Results on NuScenes [2]
- **(Sec. 5)** Result video on KITTI data

1. Details on feature aggregation

We introduce our Feature transform module (FTM) in Sec. 3.2.1 of the main paper and utilize a summation to aggregate the features from consecutive frames. Although simple, the summation is very general in a sense that it enables our model to aggregate frames from an arbitrary amount of other frames. While we demonstrate FTM with two consecutive frames in the main paper (enabling an online system), we can easily extend our model by aggregating features from multiple frames that are further away or even from future frames in offline settings. This may not be said for other operations. For instance, feature concatenation changes the internal feature dimensions (and thus network architecture) if information from more frames are available; max pooling cannot guarantee information propagation. Moreover, our FTM can be readily extended for multiple feature maps at different spatial scales, which can potentially further improve the performance.

2. Implementation details

2.1. Model learning and inference

We use ADAM [1] to minimize $\mathcal{L}_{\text{sup}}^{\text{r}}$ and estimate the parameters of our neural network. To effectively train the entire model, we first train a $basic$ model defined as follows:

$$\Theta^t = (h \circ g_j \circ g_i)(\mathbf{x}^t). \tag{1}$$

This model receives $\mathbf{x}^t$ as input and makes predictions individually on each frame. We then add an LSTM, creating a new model denoted as $blstm$

$$\Theta^t = (h \circ g_{lstm} \circ g_j)(g_i(\mathbf{x}^t), g_i(\mathbf{x}^{t-1})) \ , \tag{2}$$

	NuScenes [2]		
Method	Accu.-Bi. ↑	Accu.-Mc. ↑	MSE ↓
BEV [3]	.846	.485	.073
H-BEV+DA [3]+GM	**.877**	.496	.032
BEV-C	.856	.471	.069
BEV-J	**.872**	.486	.036
BEV-J-O	.858	**.543**	.027
+LSTM	.859	.536	**.023**
+LSTM+FTM	.863	**.547**	**.023**

Table 1: Full results on NuScenes dataset.

	NuScenes [2]	
Method	seman.↓	temp.↓
BEV [3]	1.09	1.27
H-BEV-DA [3]+GM	0.07	0.52
BEV-J-O	0.52	1.14
+LSTM+FTM	0.10	0.51

Table 2: Consisency measurements on NuScenes dataset.

which we train with pre-trained parameters for the functions h, g_j and g_i. Finally, we add the FTM and fine-tune the full model (with parameters pre-trained as above). Details of our $basic$ and $blstm$ models can be found in the experiment section of the main paper.

In training, we set the learning rate to $1e$-4 for the model $basic$, the batch size to 26, and update the model for 50k iterations. The $blstm$ model as well as our full network are then trained with a batch size of 20 for 30k iterations and a learning rate of $1e$-5.

2.2. Scene Attributes

As mentioned in paper, we exactly follow the scene parameters defined in [3]. Specifically, we describe them in more details in Tab. 3.

Discretize continuous attributes Formulating continuous attribute prediction as a regression problem in the discretized space permits multiple modes in the final prediction. This can be further leveraged by subsequent graphical models, if available, as the unary term to find feasible solutions that avoid conflicts among different attributes.

ID	Description
B1	Is the main road curved?
B2	Is the main road a one-way?
B3	Does the main road have a delimiter?
B4	Is there a delimiter between road and side walks?
B5	Does a sidewalk exist on the left of the main road?
B6	Does a sidewalk exist on the right of the main road?
B7	Does a crosswalk exist before the intersection?
B8	Does a crosswalk exist after the intersection?
B9	Does a crosswalk exist on the left side road of the intersection?
B10	Does a crosswalk exist on right side road of the intersection?
B11	Does a crosswalk exist on the main road w/o intersection?
B12	Does a left side road exist?
B13	Does a right side road exist?
B14	Does the main road end after the side roads?
M1	Number of lanes on the left of the ego-lane (maximum 6)
M2	Number of lanes on the right of the ego-lane (maximum 6)
C1	Rotation angle of the main road (*e.g.*, when car makes a turn)
C2	Width of the right side road
C3	Width of the left side road
C4	Width of a delimiter on the main road
C5	Distance to right side street
C6	Distance to left side street
C7	Distance to crosswalk on the main road without intersections
C8	Width of delimiter between main road and sidewalk
C9	Curve radius of the main road
C10-22	Lane widths ($6 \times 2 + 1$)

Table 3: The list of all our scene attributes Θ is divided into groups as in the main paper: binary Θ_{b}, multi-class Θ_{m} and continuous Θ_{c}. Each attribute is assigned an ID preceded by its group ID (B, M or C). The color of the ID indicates if manual annotation on real data exists (green). Attributes only available in simulation are marked red.

2.3. Consistency

- *Semantic consistency:* we report the average conflicts in attribute predictions. Since no constraints are enforced explicitly in models, it is likely that they output infeasible predictions, e.g. one binary variable predicts [no sideroads] but another one predicts [crosswalk on the left side of intersection]. In this case, we count a conflict if the predicted attributes are node feasible and report the average number of conflicts as our semantic consistency measurements.
- *Temporal consistency:* we also report the average number of changes in our prediction. Specifically, if the prediction for one attribute changes in consecutive frame, we count it as a change. Intuitively, a good model should have lower number in this measurement. Note that we only consider the binary and multiclass tasks in this measurement. Also, since the predictions can be consistently wrong over time, consistency itself cannot replace accuracy.

3. IoU measurement

Here, we provide more explanations on our evaluation metric. We would like to point out that the Intersection over Union (IoU) score reported in our paper is different from what has been reported in semantic segmentation literature. In short, the two semantic maps to compute IoU are both rendered from a set of parametric attributes, rather than per-pixel predicted as usually done.

Similar to [3], our IoU has two differences compared to the IoU of a typical semantic segmentation task: 1) the semantic top-view maps are obtained with a rendering function, which is highly nonlinear and cannot be directly optimized; 2) the rendered semantic top-view maps entangle all three types of attributes together and their impact on IoU varies a lot. For instance, predicting the number of lanes on the left incorrectly by one has a bigger impact than getting the distance to a crosswalk wrong by one meter.

4. Results on nuScenes

We report more results on the nuScenes [2] dataset in Tab. 1 and Tab. 2, which we could not fit into the main paper due to the space limitation. Note that compared to the SOTA model H-BEV+DA+GM from [3], we use far less human annotations. In their experiments, Wang *et al*. have 3486 and 1042 images for training and testing, respectively. While in our case, we have 1165 images for training and 348 for testing. This is because we remove those frames where the car stops, since there is no additional temporal information to utilize and thus not relevant for our investigation. More importantly, [3] utilized 50k additional synthetic images to assist training and adopted a graphical model for post processing, which we do not use in our approach.

As can be seen in Tab. 1, our proposed method outperforms the $basic$ BEV model in all metrics. This demonstrates the effectiveness of our proposed input representation as well as the LSTM/FTM module. Compared to the SOTA method H-BEV+DA+GM [3], we show that our proposed model can achieve better performance in almost all metrics, except the binary class accuracy where we are slightly worse. Note that although our binary accuracy is worse (1.4% lower), the accuracy for multi-class scene attributes is much higher than SOTA (4.9% higher). Similarly, we show in Tab. 2 that our input representation and the LSTM/FTM modules improve

the prediction consistency over the baseline significantly, and can achieve comparable performance w.r.t. SOTA.

5. Result video

We also attach a video demonstration (see supple-video.avi for more details) of our method in this supplementary. As can be seen in the video, our predictions are quite smooth and are consistent with 3D object prediction results.

References

[1] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *ICLR*, 2015.

[2] NuTonomy. The NuScenes data set. `https://www.nuscenes.org`, 2018.

[3] Ziyan Wang, Buyu Liu, Samuel Schulter, and Manmohan Chandraker. A parametric top-view representation of complex road scenes. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.