UniFusion: Unified Multi-view Fusion Transformer for Spatial-Temporal Representation in Bird's-Eye-View – Supplementary Materials

1. Overview

In this part, we provide more detailed illustration, explanation, and visualization for the following aspects: 1) A demo of the method. 2) Comparison under different computational costs. 3) The motivation of the new 160m $\times$ 100m setting; 4) The long-range fusion ability of warp-based methods. 5) Visual comparison of different methods.

2. A demo of the method

We provide a demo of the proposed BEV map segmentation method on NuScenes, which can be found in `demo.mp4`

3. Comparison under different computational costs

Although our method could support long-range temporal fusion and gains better performance, it would has a higher computational cost compared with the short-range temporal fusion methods. For fair comparison, we scale our method's computational costs to compare with BEVFormer, as shown in Fig. A. It should be note that we only scale the Transformer module which is used for fusion. All other settings like backbone, input resolution, training settings and task-specific head remain unchanged.

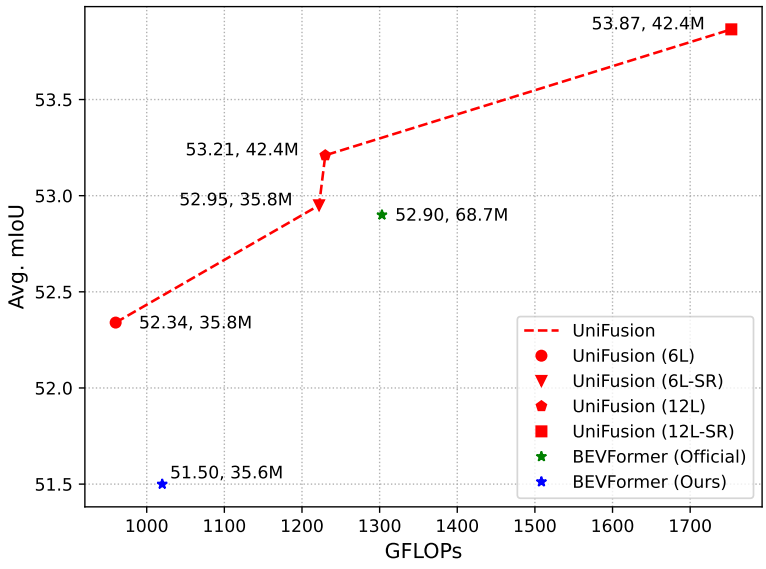

Figure A: FLOPs vs. performance. The variants of UniFusion are derived by adjusting the number of layers in the fusion Transformer and whether the self-regression is utilized. "6L" means the Transformer is 6-layer. "SR" means self-regression.

From Fig. A we can see that the proposed method could outperforms BEVFormer with lower computational costs and parameters. This shows that the proposed method could not only support long-range temporal fusion, but also has a high efficiency.

4. Motivation of the 160m $\times$ 100m setting

Generally speaking, we propose a new 160m $\times$ 100m setting that has different BEV range, line width of map element, and split compared with the existing 60m $\times$ 30m and 100m $\times$ 100m settings. The key motivations of this setting are: 1) the evaluation range should be as large as the visible limit. 2) the evaluation criterion should be discriminative for both bad and good predictions. 3) the evaluation should avoid overfitting and show the ability of generalization.

4.1. BEV Range

To determine the BEV range, we consider the visible limit of cameras. In this work, we define the visible range as the farthest point where a lane is represented by less than two pixels in the feature map (since we need to distinguish the left and right lanes of the lane, two pixels is the minimum requirement). Suppose f is the focal length of the camera, n_{pixel} is the minimal number of pixels to represent a lane, and W_{lane} is the width of the lane. The visible limit d can be written as:

$$d = \frac{f}{n_{pixel}} W_{lane} \tag{1}$$

An example of the derivation is shown in Fig. B. Typically, the focal length on NuScenes can be derived from the FOV and

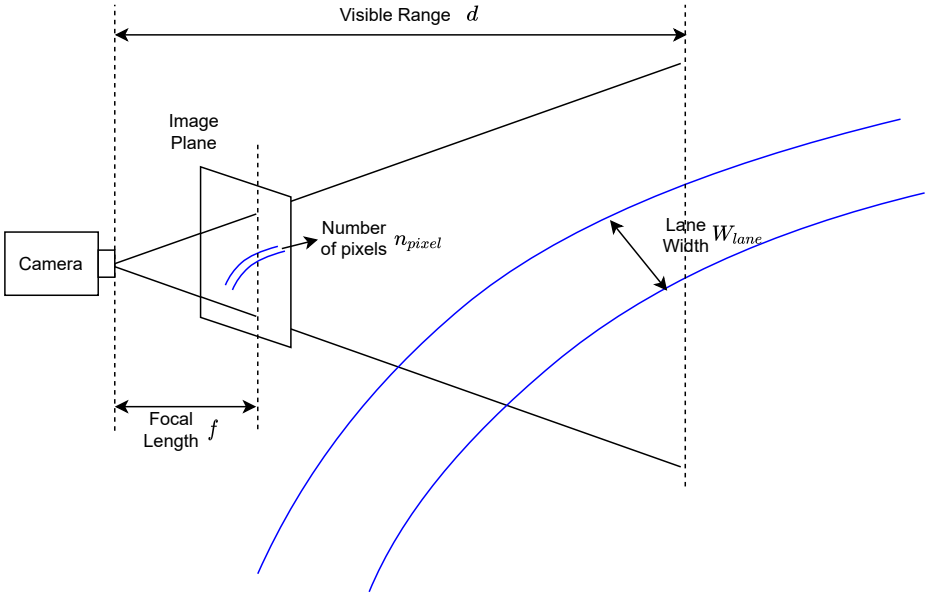

Figure B: Derivation of BEV range.

image resolution. Suppose image resolution is r, FOV is θ, and we have:

$$f = \frac{r/2}{\tan(\theta/2)} \tag{2}$$

The detailed numbers are shown in Tab. A.

Table A: The values on the NuScenes dataset. For the FOV and focal length, we list the values of front and rear cameras separately. Lane width is about 3.0m-4.0m according to the regulations of different places, and we use the minimum value of 3.0m. Since the common network output stride is larger than 32, one pixel in the feature map corresponds to at least 32 pixels in the original image.

Image Resolution r	FOV θ	Focal Lenght f	Lane Width W_{lane}	Number of pixels n_{pixel}
1600	70 / 110	1142.5 / 560.2	3.0m	32

Finally, we get the BEV range d:

$$\begin{aligned} d_{front} &= \frac{1142.5}{32} \cdot 3 \approx 107.1 \\ d_{rear} &= \frac{560.2}{32} \cdot 3 \approx 52.5 \end{aligned} \tag{3}$$

However, the rear BEV range of 52.5m is slightly short in real scenarios. We slightly extend the rear BEV range to 60m. For the left and right range, we follow the existing setting with a distance of 50m. This composes the $160\text{m} \times 100\text{m}$ setting.

4.2. Evaluation criterion

The first difference in the evaluation criterion is that all the map elements are defined as the "Line". This is because the polygon area is not suitable for representing road structures and the mIoU metric with polygon is abnormally high. For example, the "Road mIoU" is about 80 while the "Lane mIoU" is only about 20.

The second part of our evaluation is the line width. In this work, we use 3-pixel-wide lines. This is to avoid the problem of the 1-pixel evaluation. For example, if the predicted lane is only shifted by 1 pixel from the ground truth, then the mIoU is 0. There is no discrimination for "wrong but close" and 'totally wrong'" cases under this setting. This property also causes another problem, that is, if we simply upsample the ground truth and make the prediction also works in high resolution, the performance would increase significantly, which would cause an unfair comparison between different methods. To avoid these problems, we set the line width to 3 pixels. For the predictions that are close to ground truth but not exactly correct, our evaluation could also give responses to these results and are more discriminative. For the upsample problem, since we make the original 1-pixel "lane mIoU" a 3-pixel "area mIoU", the upsampled results are less affected.

4.3. City-based split

In our setting, we also propose the city-based split for NuScenes. This is because the vanilla training and validation splits in NuScenes contain many similar scenes, which potentially suffer from the overfitting problem. In this way, we propose a split that is based on the cities and locations on NuScenes. NuScenes is collected in four places, which are "singapore-onenorth", "singapore-queenstown", "singapore-hollandvillage", and "boston-seaport". We use the samples collected in "singapore-queenstown" and "singapore-hollandvillage" as the training split, and "singapore-onenorth" and "boston-seaport" as the validation split. The numbers of training and validation samples are 26,093 and 8,056, respectively. For comparison, the numbers of training and validation samples in the vanilla split are 28,130 and 6,019, respectively.

5. Visualization and Comparison

In this part, we show the visualization results on NuScenes with the $160\text{m} \times 100\text{m}$ setting. Moreover, we also show the results of other method for comparison in Fig. C. From Fig. C, we can see that our method gains the best results. The prediction of lines in our method is smooth and clear.

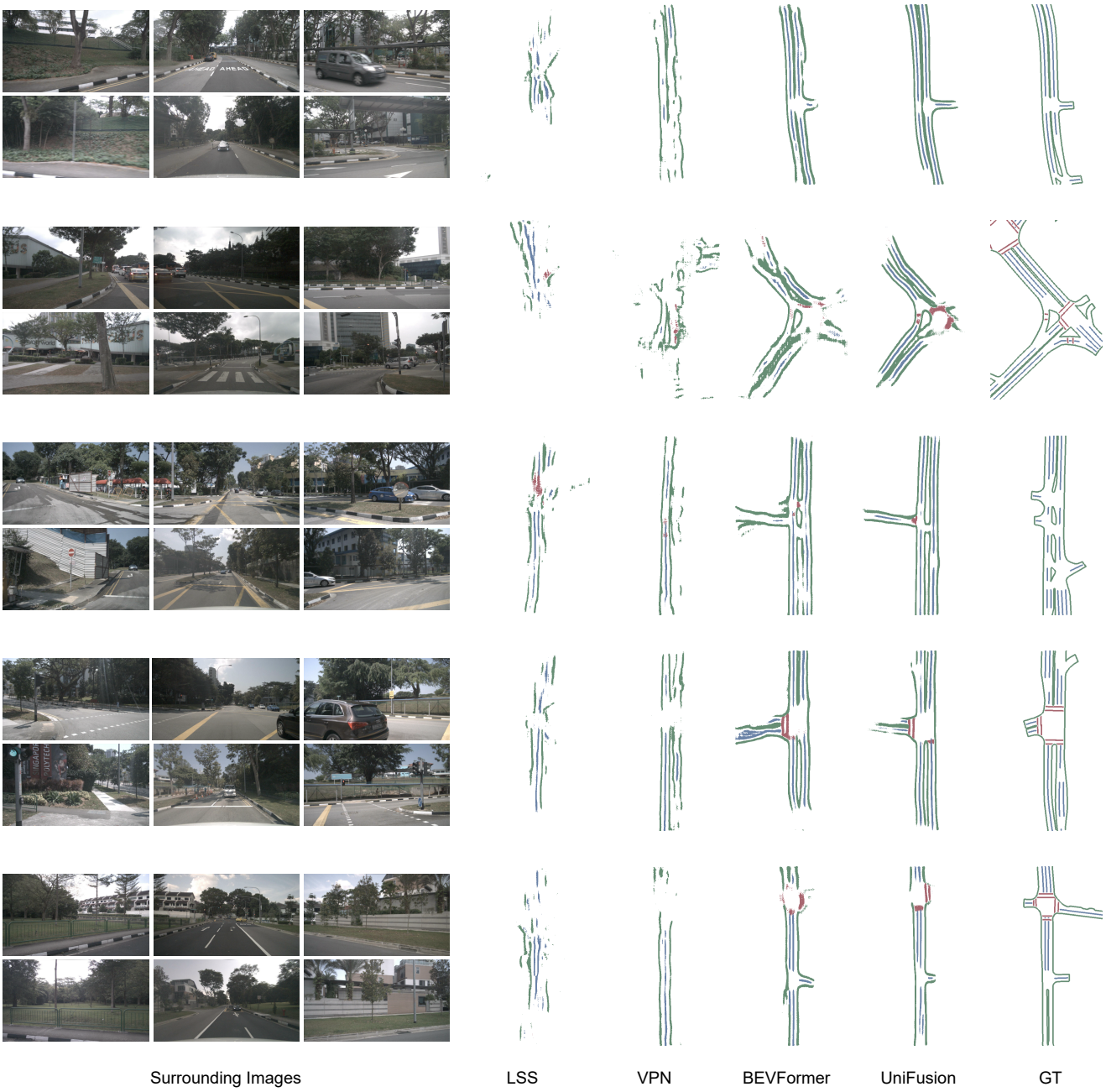

Figure C: The visual comparison on the city-based val split of NuScenes with the 160m $\times$ 100m setting. Best viewed when zoomed in.