Online Vectorized HD Map Construction using Geometry
Supplementary Material

This supplementary material is organized as follows:
- More details on the method design (§ A).
- Further quantitative experimental results (§ B).
- Additional visualization results under three weather conditions (§ C).

A Additional Details

A.1 Objective Functions

Objective Configurations. Our method employs two distinct objective functions. The full objective function is defined as follows:

$$\begin{aligned}\mathcal{L} = \lambda \cdot \mathcal{L}_{\text{Euc}} + \beta_1 \cdot \mathcal{L}_{\text{cls}} + \beta_2 \cdot \mathcal{L}_{\text{pts}} \\ + \beta_3 \cdot \mathcal{L}_{\text{dir}} + \beta_4 \cdot \mathcal{L}_{\text{seg}} + \beta_5 \cdot \mathcal{L}_{\text{dep}}\end{aligned} \tag{1}$$

and the simpler one which excludes dense prediction losses is:

$$\mathcal{L}' = \lambda \cdot \mathcal{L}_{\text{Euc}} + \beta_1 \cdot \mathcal{L}_{\text{cls}} + \beta_2 \cdot \mathcal{L}_{\text{pts}} + \beta_3 \cdot \mathcal{L}_{\text{dir}}. \tag{2}$$

Point Order Agnostic Matching. In accordance with the methodology proposed by MapTR [22], we employ point order-agnostic matching between the prediction and ground truth. In the subsequent formulations, we assume that the prediction and ground truth have already been paired.

Classification Loss. To enhance the model's comprehension of semantics associated with various map instance types, we incorporate the classification task. Let $\hat{\boldsymbol{p}} \in \mathbb{R}^{N \times C}$ denote the predicted probabilities, where C is the number of instance categories. Here, $\hat{\boldsymbol{p}}_{ic}$ represents the predicted probability of instance i belonging to category c. With ground truth labels $\boldsymbol{y} \in \{1, ..., C\}^N$, the objective function based on focal loss is defined as follows:

$$\mathcal{L}_{\text{cls}} = -\sum_{i=1}^{N}\sum_{c=1}^{C} \delta[\boldsymbol{y}_i = c] \cdot \alpha_c (1 - \hat{\boldsymbol{p}}_{ic})^{\gamma} \log \hat{\boldsymbol{p}}_{ic}\,, \tag{3}$$

where $\delta[q] = 1$ if proposition q is true and $\delta[q] = 0$ otherwise.

Point Loss. For the perception of instance positions, we employ a point loss that evaluates L^1 distances between predicted points and ground truth points, which is specified as:

$$\mathcal{L}_{\text{pts}} = \sum_{i=1}^{N}\sum_{j=1}^{N_v} \|\hat{\boldsymbol{L}}_j^i - \boldsymbol{L}_j^i\|_1. \tag{4}$$

Edge Direction Loss. To obtain more precise displacement vectors, which are crucial in our G-Representation, we incorporate an edge direction loss. This loss

quantifies the cosine similarity between predicted displacement vectors and their corresponding ground truth vectors. Specifically, the loss is defined as:

$$\mathcal{L}_{\text{dir}} = -\sum_{i=1}^{N}\sum_{j=1}^{N_v} \frac{(\hat{\boldsymbol{v}}_j^i)^\top \boldsymbol{v}_j^i}{\|\hat{\boldsymbol{v}}_j^i\|_2 \cdot \|\boldsymbol{v}_j^i\|_2}. \tag{5}$$

Segmentation Loss. The auxiliary binary segmentation task is valuable for assisting the model in the coarse perception of shape geometry. We integrate a convolutional neural network-based BEV segmentation head with BEV features. Let $\hat{\boldsymbol{P}}_{\text{bev}} \in \mathbb{R}^{H' \times W'}$ represent the probability of each grid belonging to the instance area, and $\boldsymbol{Y}_{\text{bev}} \in \{0,1\}^{H' \times W'}$ denote the ground truth. The corresponding objective function is defined as:

$$\mathcal{L}_{\text{bev}} = \mathcal{L}_{\text{bce}}(\hat{\boldsymbol{P}}_{\text{bev}}, \boldsymbol{Y}_{\text{bev}}), \tag{6}$$

where the binary cross entropy loss $\mathcal{L}_{\text{bce}}$ is:

$$\begin{aligned} \mathcal{L}_{\text{bce}}(\hat{p}, y) = &- \delta[y = 1] \cdot \log \hat{p} \\ &- \delta[y = 0] \cdot \log(1 - \hat{p}). \end{aligned} \tag{7}$$

We also introduce the auxiliary PV segmentation task, incorporating a shared convolutional neural network head for all views. The ground truth is projected back to the PV space to form the binary mask. Let $\hat{\boldsymbol{P}}_{\text{pv}}^k \in \mathbb{R}^{H \times W}$ denote the segmentation results for view k with corresponding ground truth $\boldsymbol{Y}_{\text{pv}}^k \in \{0,1\}^{H \times W}$, then the objective function can be expressed as:

$$\mathcal{L}_{\text{pv}} = \sum_{k=1}^{K} \mathcal{L}_{\text{bce}}(\hat{\boldsymbol{P}}_{\text{pv}}^k, \boldsymbol{Y}_{\text{pv}}^k). \tag{8}$$

Finally, we obtain the segmentation loss as follows:

$$\mathcal{L}_{\text{seg}} = \beta_{\text{bev}} \cdot \mathcal{L}_{\text{bev}} + \beta_{\text{pv}} \cdot \mathcal{L}_{\text{pv}}. \tag{9}$$

Depth Estimation Loss. To enhance depth perception, we adopt an auxiliary depth estimation task. Let $\hat{\boldsymbol{P}}_{\text{dep}}^k \in \mathbb{R}^{H \times W \times D}$ represent the depth distribution of each grid estimated by LSS [35] in the PV space of view k, where D represents the number of quantified depth buckets. Given the ground truth $\boldsymbol{Y}_{\text{dep}}^k \in \{1, ..., D\}^{H \times W \times D}$, the depth estimation loss is defined as:

$$\mathcal{L}_{\text{dep}} = -\sum_{k=1}^{K}\sum_{d=1}^{D} \delta[\boldsymbol{Y}_{\text{dep}}^k = d] \cdot \log \hat{\boldsymbol{P}}_{\text{dep}}^k. \tag{10}$$

A.2 Hyperparameter Settings

In the default optimization setting, we set the dropout rate to 0.1 and weight decay to 0.03. The first 500 iterations involve a linear warm-up, starting from 1/3

of the maximum learning rate. In the Cosine Annealing scheduler, the minimum learning rate is set to 0.001 of the maximum. Unless explicitly stated otherwise, we train our model for 110 epochs on nuScenes and 24 epochs on Argoverse 2. For the simplified objective configuration, we set the maximum learning rate to 6×10^{-4} with a batch size of 4. When LiDAR input is utilized, the batch size is reduced to 3. In the full objective configuration, varied hyperparameters are detailed in Table A2. Also, the default hyperparameter settings for objective functions are presented in Table A1.

Table A1: Hyperparameters of objective functions.

Parameter	α_c	γ	λ	β_1	β_2
Value	0.25	2	0.005	2	5
Parameter	β_3	β_4	β_5	β_{bev}	β_{pv}
Value	0.005	1	3	1	2

Table A2: Hyperparameters under different vision backbones.

Backbone	Max Learning Rate	Batchsize
R50	6×10^{-4}	4
V2-99	6×10^{-4}	3
Swin-T	4×10^{-4}	3

Moreover, we set the number of instance queries as $N = 50$ and the number of point queries as $N_v = 20$. We employ a single layer of encoder in GKT and incorporate 6 attention blocks in the Geometry-Decoupled Decoder. In the context of LSS transformation, the depth spans from 1 to 35 meters, quantified at intervals of 0.5 meters, resulting in $D = 68$.

B More Experimental Results

In this section, we present additional ablation studies and hyperparameter experiment results. In all of these experiments, the model is trained for 24 epochs on nuScenes using the simplified objective function. Unless otherwise specified, we employ the default settings outlined in § A.2.

B.1 Impact of the Decoder Block Number

We evaluate the impact of decoder block numbers on the model performance, as presented in Table A3. When increasing the number of blocks from 1 to 6,

the mAP increases by +20.8%. However, naively adding more blocks might be detrimental to model performance. For example, mAP decreases by −4.7% when increasing the number of blocks from 6 to 12.

Table A3: Impact of the decoder block number. The default setting utilized in our experiments is highlighted in gray.

# Block	AP_{div}(↑)	AP_{ped}(↑)	AP_{bnd}(↑)	mAP(↑)
1	33.5	24.7	37.3	31.8
2	42.1	38.9	48.2	43.1
4	51.1	43.5	53.9	49.5
6	53.6	**49.2**	**54.8**	**52.6**
8	**54.5**	46.4	53.4	51.4
10	52.4	45.7	53.5	50.5
12	49.6	45.1	48.9	47.9

B.2 Impact of the Query Number

We also evaluate the influence of query numbers on model performance, as detailed in Table A4 for instance queries and Table A5 for point queries.

Instance Queries. As depicted in Table A4, augmenting the number of instance queries could be advantageous for the model's performance. More specifically, the mAP exhibits an increment of +27.8% when the query number is elevated from 10 to 50. This observation aligns with intuition, as a higher number of instance queries implies a broader pool of diverse candidates.

Point Queries. It is observed from Table A5 that an excess or insufficient number of point queries has an adverse impact on the model performance. Notably, an interesting finding is that the optimal query number varies according to different instance categories. For example, lane dividers exhibit better performance with $N_v = 10$, while pedestrian crossings and road boundaries show optimal results with $N_v = 20$. This discrepancy is attributed to the straight shape of lane dividers, whereas pedestrian crossings and road boundaries, characterized by more intricate shapes, benefit from a relatively larger point query number. Hence, the results suggest that adapting point query numbers based on the complexity of instance geometry could further enhance the model performance, which is a topic left for future investigation.

C More Visualization Results

We present additional visualization cases under varied weather conditions, as illustrated in Figure A1 to Figure A3. Our method is trained with a ResNet50 backbone using the simplified objective function.

As illustrated in Figure A1, in challenging rainy conditions, our method demonstrates more robust results. Particularly in scenario (d) of Figure A1,

Table A4: Impact of the instance query number.

N	$\text{AP}_{div}(\uparrow)$	$\text{AP}_{ped}(\uparrow)$	$\text{AP}_{bnd}(\uparrow)$	mAP(↑)
10	30.2	12.3	31.9	24.8
30	50.6	43.4	50.5	48.2
40	51.0	47.5	53.1	50.5
50	**53.6**	**49.2**	54.8	**52.6**
60	52.6	49.0	**55.6**	52.4

Table A5: Impact of the point query number.

N_v	$\text{AP}_{div}(\uparrow)$	$\text{AP}_{ped}(\uparrow)$	$\text{AP}_{bnd}(\uparrow)$	mAP(↑)
5	49.7	31.4	41.8	41.0
10	**53.7**	45.9	52.5	50.7
20	53.6	**49.2**	**54.8**	**52.6**
30	50.9	48.3	54.7	51.3
40	50.2	47.9	54.6	50.9

where the front road boundary and lane divider are heavily occluded by water on the front windshield, our method can still recover the entire instance accurately from observed parts. This showcases the potential of proposed geometric designs.

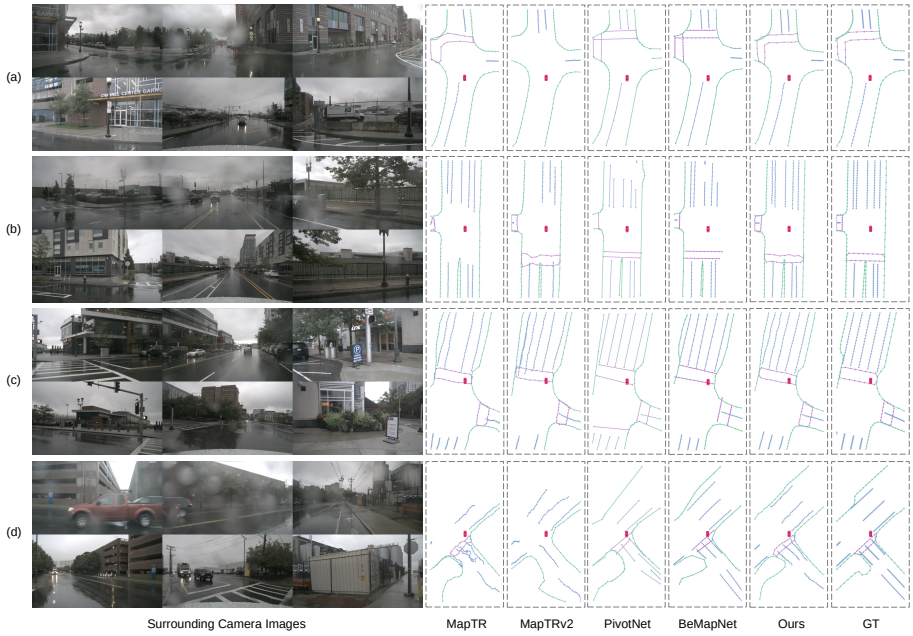

Fig. A1: Visualization results under challenging rainy weather conditions. Even with noisy reflections on the road and map instances occluded by water drops, our method still provides robust predictions.

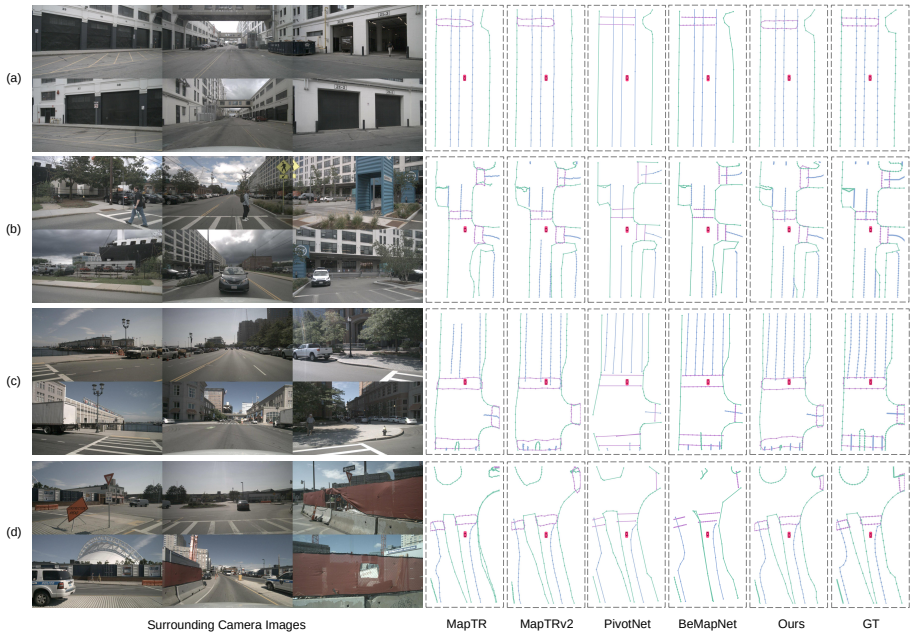

Fig. A2: Visualization results under sunny weather conditions.

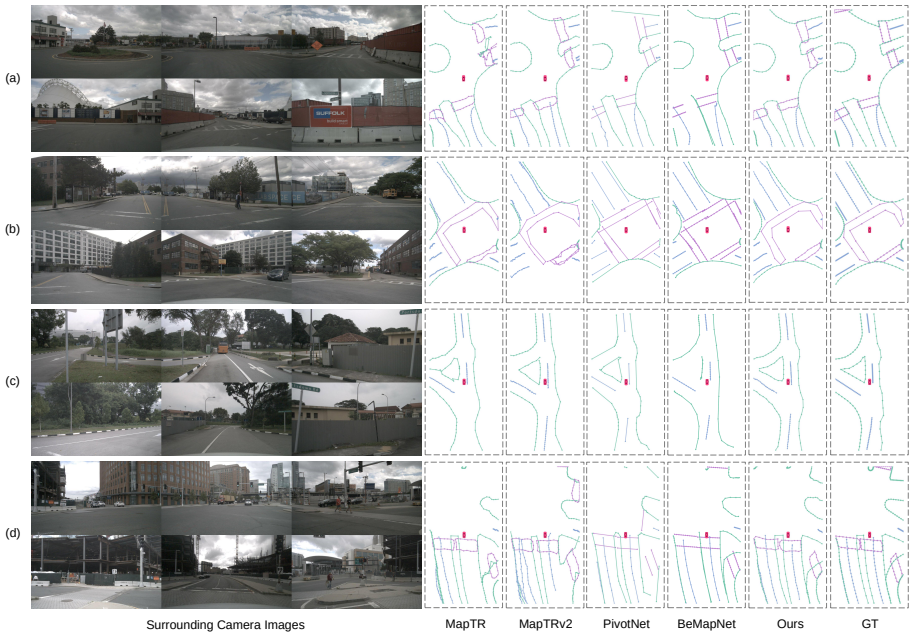

Fig. A3: Visualization results under cloudy weather conditions.