CLIP-BEVFormer: Enhancing Multi-View Image-Based BEV Detector with Ground Truth Flow

Supplementary Material

1. Experiment Setup

In our experiments, we adopt ResNet50 and ResNet101 [2] as backbones for the BEVFormer-tiny and BEVFormer-base models, respectively. These backbones are initialized from the FCOS3D [6] checkpoint, following the configuration in BEVFormer [3]. We leverage the output multi-scale features from the Feature Pyramid Network (FPN) [4], with sizes of 1/16, 1/32, and 1/64, and the dimension of 256. During the training phase for CLIP-BEVFormer, we leverage the pretrained language model in CLIP-RN101 [5] as our off-the-shelf language model.

The BEV size for the tiny and base variants is set to 50×50 and 200×200, respectively, while the perception ranges span from -51.2m to 51.2m along the X and Y axes. The resolution of the BEV grid is set to 0.512m. We incorporate learnable positional embeddings for BEV queries to enrich the spatial representation.

The BEV encoder comprises 6 encoder layers, consistently refining BEV queries in each layer. During the spatial cross-attention module, implemented using the deformable attention mechanism, each local query corresponds to four target points with different heights in 3D space. The predefined height anchors are uniformly sampled from -5 meters to 3 meters.

For each reference point on 2D view features, we utilize four sampling points around this reference point for each head. During training, we use a 2-frame history BEV for the tiny variant and a 3-frame history BEV for the base variant. We train our models for 24 epochs with a learning rate of 2×10^{-4} [3].

2. 3D Object Detection Results with Various Baselines

We have conducted experiments with various detection baselines, BEVformer [3], BEVformerV2 [7], and BEVerse [8]. We evaluate our model on both validation and test sets of nuScenes. The results presented in Tab. 1 show that our method consistently improves the perception capabilities of various baselines by significant margins on both sets, indicating its flexibility and model-agnostic nature.

3. 3D Object Detection Metrics

We adhere to standard evaluation metrics for 3D detection on the nuScenes dataset [1], encompassing metrics such as mean Average Precision (mAP), Average Translation Error (ATE), Average Scale Error (ASE), Average Orientation Error (AOE), Average Velocity Error (AVE), Average Attribute Error (AAE), and nuScenes detection score (NDS).

Mean Average Precision (mAP). For mAP, we utilize the Average Precision metric, modifying the definition of a match by considering the 2D center distance on the ground plane instead of intersection over union-based affinities. Specifically, we match predictions with ground truth objects based on the smallest center distance within a certain threshold. Average precision (AP) is calculated by integrating the recall vs precision curve for recalls and precisions > 0.1. We then average over match thresholds of 0.5, 1, 2, 4 meters and compute the mean across classes.

True Positives (TP). TP metrics are designed to measure translation, scale, orientation, velocity, and attribute errors. These are calculated using a threshold of 2m center distance during matching and are positive scalars. Metrics are defined per class, and we then take the mean over classes to calculate mATE, mASE, mAOE, mAVE, and mAAE.

- **Average Translation Error (ATE).** Euclidean center distance in 2D in meters.
- **Average Scale Error (ASE).** Calculated as 1 - IOU after aligning centers and orientation.
- **Average Orientation Error (AOE).** Smallest yaw angle difference between prediction and ground truth in radians. Orientation error is evaluated at 360 degrees for most classes, except barriers, where it is evaluated at 180 degrees. Orientation errors for cones are ignored.
- **Average Velocity Error (AVE).** Absolute velocity error in m/s. Velocity error for barriers and cones is ignored.
- **Average Attribute Error (AAE).** Calculated as 1 - acc, where acc is the attribute classification accuracy. Attribute error for barriers and cones is ignored.

nuScenes Detection Score (NDS). We consolidate the above metrics by computing a weighted sum: mAP, mATE, mASE, mAOE, mAVE, and mAAE. As a first step, we convert TP errors to TP scores using TP_score = max(1 - TP_error, 0.0). We then assign a weight of 5 to mAP and 1 to each of the 5 TP scores, calculating the normalized sum.

4. Training and Inference Efficiency

Our model is trained with 4 A100 80GB GPUs. Our method does not introduce any additional parameters and computations during the inference stage, which means that it allows for enhanced performance without sacrificing real-time processing capabilities. We provide detailed information on memory, training time, number of parameters and FPS in Tab. 2.

5. Visualization

In Fig. 1 and Fig. 2, we present a comprehensive visualization of the qualitative detection performance achieved by CLIP-BEVFormer. The images provide insights into both camera and Bird's Eye View (BEV) perspectives, offering a nuanced understanding of the model's predictions. Notably, these visualizations highlight the enhanced alignment between CLIP-BEVFormer's predictions and ground truth detections in both camera and BEV views, underscoring the model's proficiency in accurately capturing the 3D environment.

References

[1] 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. 1

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[3] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *European conference on computer vision*, pages 1–18. Springer, 2022. 1

[4] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 1

[5] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 1

[6] Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. Fcos3d: Fully convolutional one-stage monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 913–922, 2021. 1

[7] Chenyu Yang, Yuntao Chen, Hao Tian, Chenxin Tao, Xizhou Zhu, Zhaoxiang Zhang, Gao Huang, Hongyang Li, Yu Qiao, Lewei Lu, et al. Bevformer v2: Adapting modern image backbones to bird's-eye-view recognition via perspective supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17830–17839, 2023. 1

[8] Yunpeng Zhang, Zheng Zhu, Wenzhao Zheng, Junjie Huang, Guan Huang, Jie Zhou, and Jiwen Lu. Beverse: Unified perception and prediction in birds-eye-view for vision-centric autonomous driving. *arXiv preprint arXiv:2205.09743*, 2022. 1

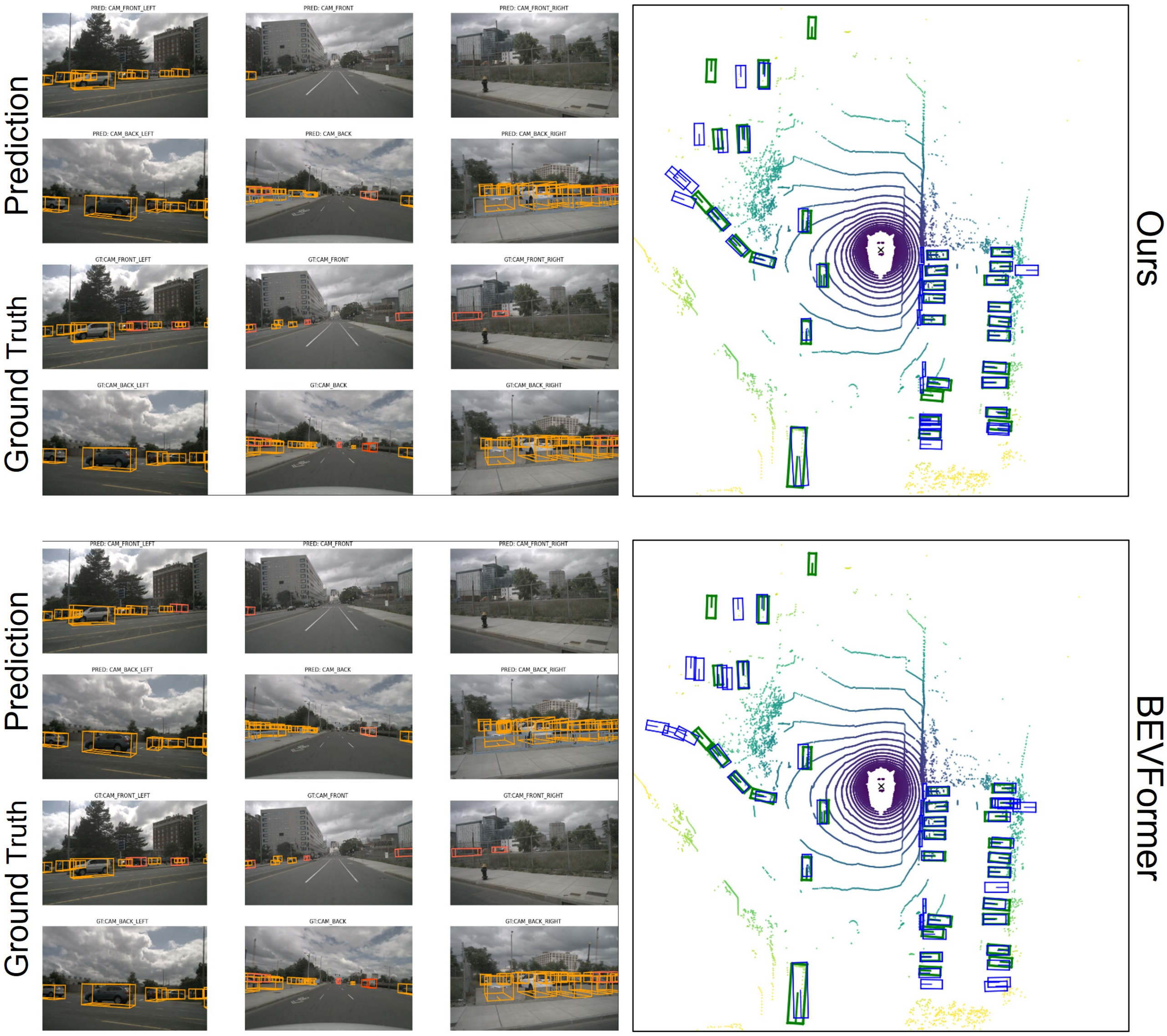

Figure 1. **Visualization results on nuScenes validation set.** We demonstrate qualitative detection performance on both camera and BEV images. As can be seen in BEV images, our CLIP-BEVFormer method demonstrates improved alignment with ground truth detections.

Model	Backbone	Validation Set							Test Set						
		NDS ↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓	NDS ↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
BEVformer-tiny	R50	35.5	25.1	0.898	0.293	0.651	0.657	**0.216**	37.2	27.3	0.856	0.283	0.609	0.753	0.146
+Ours	R50	**38.8**	**27.3**	**0.856**	**0.282**	**0.583**	**0.538**	0.228	**41.1**	**29.3**	**0.811**	**0.271**	**0.554**	**0.579**	**0.136**
BEVformer-base	R101	51.7	41.6	0.673	0.274	0.372	0.394	0.198	53.5	44.5	0.631	0.257	**0.405**	0.435	0.143
+Ours	R101	**55.1**	**44.1**	**0.641**	**0.253**	**0.319**	**0.307**	**0.172**	**54.7**	**44.7**	**0.591**	**0.257**	0.417	**0.371**	**0.128**
BEVformerV2	R50	42.6	35.1	0.753	0.286	0.466	0.807	**0.186**	42.5	35.4	0.707	0.278	0.506	**0.895**	**0.134**
+Ours	R50	**44.1**	**37.0**	**0.729**	**0.281**	**0.438**	**0.791**	0.204	**43.6**	**37.9**	**0.676**	**0.272**	**0.475**	0.975	0.141
BEVerse	Swin	46.6	32.1	0.681	0.278	0.466	0.328	0.190	50.1	36.2	0.610	0.257	0.451	0.355	0.131
+Ours	Swin	**48.3**	**34.2**	**0.665**	**0.270**	**0.456**	**0.318**	**0.170**	**52.2**	**37.4**	**0.556**	**0.247**	**0.413**	**0.301**	**0.129**

Table 1. 3D object detection results on nuScenes validation and test sets.

CVPR
#8579

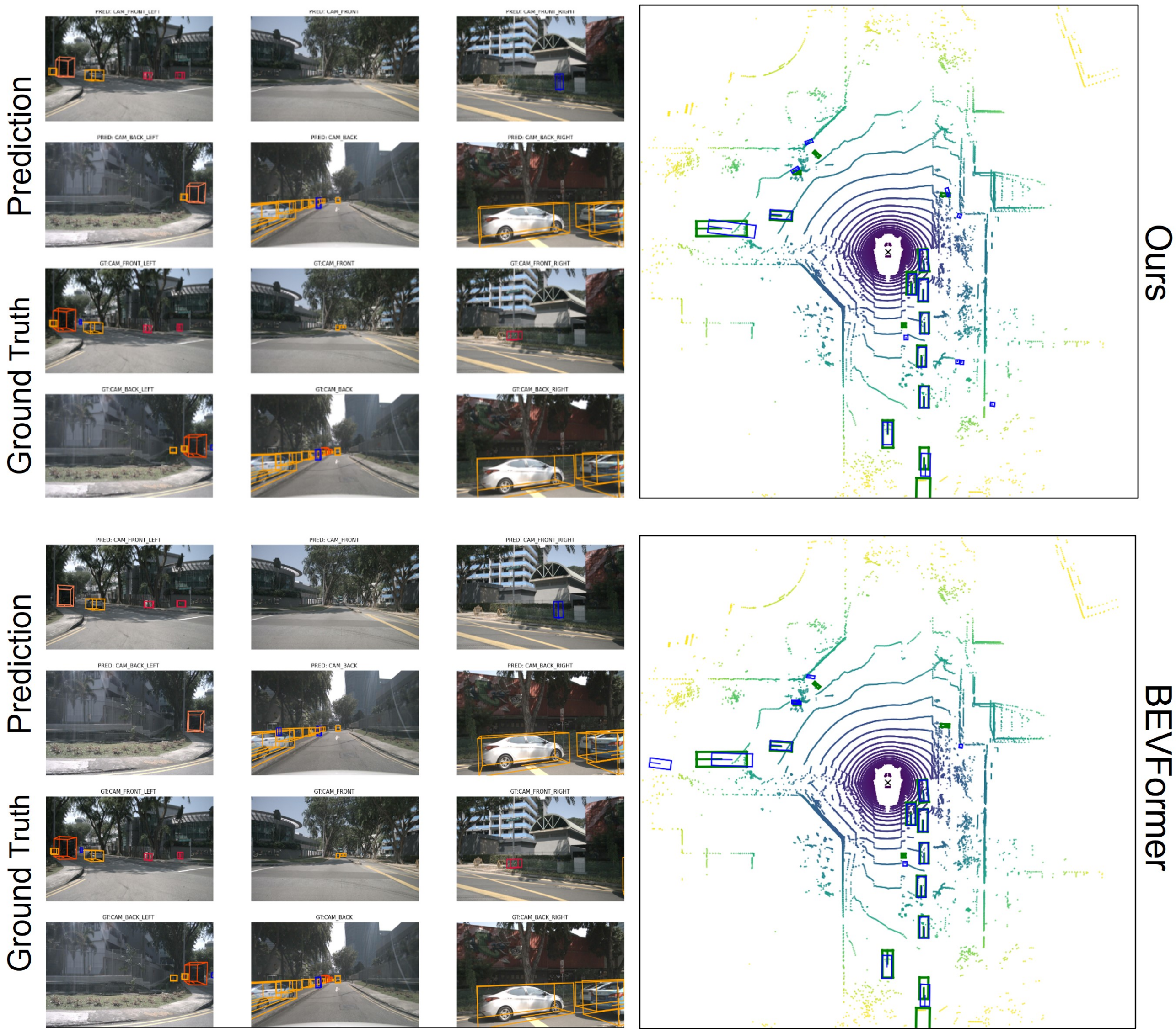

Figure 2. **Visualization results on nuScenes validation set.** Our CLIP-BEVFormer demonstrates improved alignment with ground truth detections on both camera and BEV images.

CVPR
#8579

Model	Train Mem (GB)	Train Hrs	# Params (M)	FPS
BEVformer-tiny	$\sim$7	$\sim$46	33	5.1
+Ours	$\sim$7	$\sim$46	33	5.1
BEVformer-base	$\sim$25	$\sim$90	69	2.1
+Ours	$\sim$25	$\sim$90	69	2.1
BEVformerV2	$\sim$46	$\sim$38	56	2.3
+Ours	$\sim$46	$\sim$38	56	2.3
BEVerse	$\sim$48	$\sim$72	102.5	4.4
+Ours	$\sim$48	$\sim$72	102.5	4.4

Table 2. Efficiency details. FPS is tested on 1 V100 GPU.