3D Occupancy and Flow Prediction based on Forward View Transformation

Yun Zhao, Peiru Zheng, Zhan Gong
IEIT Systems
{zhaoyun02, zhengpeiru, gongzhan01}@ieisystem.com

Abstract

In this technical report, we summarize our solution for the 'Occupancy and Flow' track of the Autonomous Grand Challenge, which is held in conjunction with the CVPR 2024 Workshop on Foundation Models for Autonomous Systems. The proposed method utilizes a forward projection-based 3D perception framework, wherein the image features are projected into the egocentric coordinate system based on the camera extrinsics and intrinsics. On the basis of the forward framework, we design an improved 3D encoder module and propose several optimization methods for the 3D occupancy and flow prediction task, including the generation of the visible mask. Our proposed method achieves a state-of-the-art occupancy score of 0.489 on the nuScenes OpenOcc dataset in this challenge track.

1. Introduction

3D surrounding perception is an essential task within the autonomous driving perception system, providing rich information for planning tasks. Instead of using a 3D bounding box to represent the 3D object , the 3D occupancy perception task involves recognizing the fine structure of the object and representing the complex scene background. A typical representation of 3D occupancy involves voxelizing the 3D space and labeling each voxel with its occupancy state and semantics. In this challenge, we aim to estimate the semantics of each voxel grid and predict the flow of the foreground voxels using only surround-view images.

Based on the view transformation pattern, current camera-based 3D perception methods can be categorized into three main types: 1) Forward projection [2, 10], which lifts 2D image features into the 3D volume via estimated depth. 2) Backward projection [4, 5, 8], which constructs 3D queries in the volume and retrieves 2D image features using a cross-attention mechanism. 3) Fusion of forward and backward projection techniques [7]. Our proposed method employs the forward projection framework and leverages the long-term previous information [9] to enhance the feature representation in the current frame.

2. Our Solution

2.1. Model Structure

The pipeline of our method is shown in Fig. 1. The overall architecture consists of four main components: an image-based encoder, a forward transformation module, a 3D feature encoder, and a prediction head.

Firstly, the surrounding images are encoded into multiple feature maps through the shared image-based encoder, which includes a pre-trained image backbone and an FPN module. In the forward transformation module, we follow the LSS [10] to project the image features into a unified 3D space based on the estimated depth of each pixel. Additionally, we adopt the approach from BEVDepth [3] to refine the depth estimation network and supervise the depth training process using the depth map generated from the point clouds. The projected features are voxelized into 3D volume features and fed into the 3D feature encoder. Finally, the prediction employs the 3D volume features to predict the semantics and flow of each voxel.

Next, we will introduce the selection of the 2D image backbone and the modifications made to the 3D-based feature encoder in our solution.

2.1.1 2D Backbone

To improve model accuracy, a convenient and effective approach is scaling the model size. However, a larger number of parameters increases computational costs and requires more data to train the model. Considering both effectiveness and efficiency, we select FlashInternImage [12] as our 2D image backbone. FlashInternImage [12] introduces a core operator, DCNv4, which is modified based on the DCNv3 in InternImage [11], to enhance efficiency and effectiveness. In our experiments, the methods using the FlashInternImage-T ($\sim$30M parameters) demonstrate impressive performances. Ultimately, we employ the FlashInternImage-L [12] ($\sim$220M parameters) to achieve a better performance.

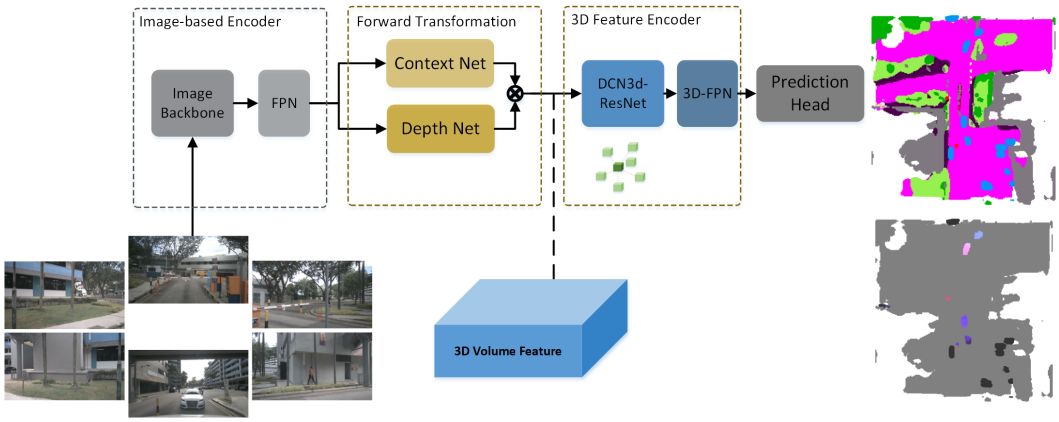

Figure 1. Overall architecture. The view transformation is based on the forward projection strategy (BEVDepth [3]). The 3D feature encoder utilizes the deformable 3D convolution core. Prediction results encompass the occupancy semantics and flow. The color in the flow map indicates voxel orientation and the intensity reflects the velocity magnitude. Background voxels are depicted in gray to distinguish them from foreground objects.

2.1.2 DCN3D

To enhance the expressiveness of the 3D volume features, we replace the traditional 3D convolution operator with the deformable 3D convolution (DCN3D) in the 3D-based feature encoder. Increasing the 3D volume size can improve prediction accuracy by focusing on detailed environmental information. However, this can lead to a significant increase in computational and memory consumption. To mitigate the substantial computational and memory demands, we modify the DCN3D operator following the DCNv4 [12] by optimizing memory access to minimize redundant operations. The DCN3D kernel is implemented in CUDA to conserve computational resources and memory.

2.2. Training

2.2.1 Data Processing: Visible Mask Generation

The provided occupancy dataset contains many invisible grids in the camera observation, which can interfere with model training. We generate a visible mask for the training subset by referring to the ray casting approach used in SparseOcc [8]. Initially, query rays are projected into the 3D ground-truth occupancy volume and are halted if they intersect with any surface. Rays that do not intersect with any occupancy voxel are excluded. Subsequently, we label all voxels along the remaining line segment as visible and label the rest as invisible. Furthermore, to mitigate the illusory prediction of objects beyond the given perception range, we randomly designate a certain proportion of invisible grids near the volume edges as visible.

2.2.2 Loss

For occupancy prediction, we utilize the Focal loss, Affinity Loss, and Lovasz-Softmax Loss, following the FB-OCC [6]. For flow prediction, we use the L1Loss and assign the voxel with the weight based on the norm value of their speed. Additionally, we incorporate the depth supervision loss.

2.3. Post-Process

Flow evaluation is conducted on 8 types of objects. During the inference phase, we set the speed of voxels estimated as background or free space to zero. We do not employ test-time augmentation or the model ensemble methods.

3. Experiments

3.1. Datasets and Metrics

Dataset. The challenge occupancy dataset is built based on the nuScenes dataset [1], which contains 17 classes. The velocity is labeled for the 8 classes ('car', 'truck', 'trailer', 'bus', 'construction vehicle', 'bicycle', 'motorcycle', 'pedestrian') in m/s. The range of the occupancy is [-40m, -40m, -1m, 40m, 40m, 5.4m]. The dataset contains 28130/6019/6008 frames for training/validation/testing.

Metrics. The challenge evaluation metric is the occupancy score, which consists of Ray-based mIoU [8] and the absolute velocity error for occupancy flow. The ray-based mIoU performs mean intersection-over-union on the query rays under different distance thresholds. For the occupancy flow, it measures velocity errors for the true positive detections of the 8 classes under a threshold of 2m distance. The final occupancy score is defined as follows: $\text{OccScore} = \text{mIoU} * 0.9 + \max(1 - \text{mAVE}, 0) * 0.1$.

Method	Framew.	Image Res.	Voxel S.	Im. B.	Occ score	RayIoU	mAVE	RayI.@1	RayI.@2	RayI.@4
Baseline	BevF. [5]	900 × 1600	0.4	R.-50	0.355	0.285	0.019	0.222	0.291	0.343
Version A	BevF. [5]	900 × 1600	0.4	R.-50	0.372	0.305	0.025	0.242	0.312	0.362
Version B	FB-Occ [6]	256 × 704	0.8	R.-50	0.320	0.335	0.816	0.276	0.342	0.387
Version C	FB-Occ [6]	256 × 704	0.8	R.-50	0.340	0.346	0.707	0.283	0.354	0.400
Version D	FB-Occ [6]	256 × 704	0.8	R.-50	0.352	0.353	0.652	0.290	0.361	0.407
Version E	F-Occ	256 × 704	0.8	Flash.-T	0.387	0.385	0.601	0.322	0.394	0.440
Version F	F-Occ	256 × 704	0.8	Flash.-L	0.404	0.403	0.589	0.342	0.413	0.455
Version G	F-Occ	640 × 1600	0.8	Flash.-T	0.410	0.410	0.587	0.345	0.421	0.463
Version H	F-Occ	256 × 704	0.4	Flash.-T	0.410	0.399	0.493	0.339	0.407	0.452

Table 1. 3D occupancy and flow prediction performance of different settings.

3.2. Implementation Details

The baseline model is the official BEVFormer [5]. We implement our solution, modifying the FB-OCC [6] by eliminating the backward projection module. During training, we use image data augmentation techniques, including random flipping, scaling and rotation. We use the AdamW optimizer and employ a one-cycle learning rate strategy. We use 16 historical frames in online mode during training following SOLOFusion [9].

For the final submission, we set the image resolution to 640×1600 and train the model based on FlashInternImage-L for 24 epochs. The estimated depth is in 110 discrete categories from 2m to 46m, and the resolution of the 3D volume voxel is $200 \times 200 \times 16$. The submitted model is trained on 8 GPUs with a global batch size of 16.

3.3. Ablation

During exploration, we implement the solution in different settings. The 3D occupancy and flow prediction performance of different settings are shown in Tab. 1.

The baseline method is the official implementation based on the BEVFormer framework. During training, we randomly eliminate 80% of the voxels labeled as free to address the imbalance between the foreground and background. In version A, we generate the visible mask and randomly eliminate 80% of the visible free voxels. Since we perform the experiments on the first version of the OpenOcc dataset, the evaluation of flow prediction may be inaccurate.

In version B, we implement the solution using the FB-Occ framework with 16 previous frames. We ignore the invisible voxels during the training phase. In version C, we improve the generation of visible mask by introducing some invisible grids near the volume edges, enhancing the model's training. For version D, we replace the custom 3D convolution in the 3D feature encoder module with the DCN3D operator.

We implement our final solution in the forward projection framework with the DCN3D-based 3D feature encoder. In version E, we use the FlashInternImage-T as the image backbone. For version F, we change the image backbone from the one in version E to the FlashInternImage-L. For version G and H, we change the image size and the voxel size from those in version E, respectively. Obviously, a large-scale model and a fine representation size can significantly improve the occupancy and flow prediction accuracy.

In our final submission, the model utilizes the F-Occ framework with the FlashInternImage-L as its image backbone. The image resolution is set to 640×1600, and the voxel size is 0.4. During inference, we disregard the flow predictions for background voxels (label > 7), assigning them a value of 0. Our approach does not incorporate test-time augmentation or model ensembling. The proposed model achieves an OccScore of 0.489 on the test set.

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. 2

[2] Junjie Huang and Guan Huang. Bevdet4d: Exploit temporal cues in multi-camera 3d object detection. *arXiv preprint arXiv:2203.17054*, 2022. 1

[3] Yinhao Li, Zheng Ge, Guanyi Yu, Jinrong Yang, Zengran Wang, Yukang Shi, Jianjian Sun, and Zeming Li. Bevdepth: Acquisition of reliable depth for multi-view 3d object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1477–1485, 2023. 1, 2

[4] Yiming Li, Zhiding Yu, Christopher Choy, Chaowei Xiao, Jose M Alvarez, Sanja Fidler, Chen Feng, and Anima Anandkumar. Voxformer: Sparse voxel transformer for camera-based 3d semantic scene completion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9087–9098, 2023. 1

[5] 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, 3

[6] Zhiqi Li, Zhiding Yu, David Austin, Mingsheng Fang, Shiyi Lan, Jan Kautz, and Jose M Alvarez. Fb-occ: 3d occupancy prediction based on forward-backward view transformation. *arXiv preprint arXiv:2307.01492*, 2023. 2, 3

[7] Zhiqi Li, Zhiding Yu, Wenhai Wang, Anima Anandkumar, Tong Lu, and Jose M Alvarez. Fb-bev: Bev representation from forward-backward view transformations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6919–6928, 2023. 1

[8] Haisong Liu, Yang Chen, Haiguang Wang, Zetong Yang, Tianyu Li, Jia Zeng, Li Chen, Hongyang Li, and Limin Wang. Fully Sparse 3D Occupancy Prediction. *arXiv e-prints*, art. arXiv:2312.17118, 2023. 1, 2

[9] Jinhyung Park, Chenfeng Xu, Shijia Yang, Kurt Keutzer, Kris M Kitani, Masayoshi Tomizuka, and Wei Zhan. Time will tell: New outlooks and a baseline for temporal multi-view 3d object detection. In *The Eleventh International Conference on Learning Representations*, 2022. 1, 3

[10] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, pages 194–210. Springer, 2020. 1

[11] Wenhai Wang, Jifeng Dai, Zhe Chen, Zhenhang Huang, Zhiqi Li, Xizhou Zhu, Xiaowei Hu, Tong Lu, Lewei Lu, Hongsheng Li, et al. Internimage: Exploring large-scale vision foundation models with deformable convolutions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14408–14419, 2023. 1

[12] Yuwen Xiong, Zhiqi Li, Yuntao Chen, Feng Wang, Xizhou Zhu, Jiapeng Luo, Wenhai Wang, Tong Lu, Hongsheng Li, Yu Qiao, et al. Efficient deformable convnets: Rethinking dynamic and sparse operator for vision applications. *arXiv preprint arXiv:2401.06197*, 2024. 1, 2