VOXELFORMER: MULTI-MODAL SPATIAL-TEMPORAL REPRESENTATION LEARNING FOR 3D OBJECT DETECTION

Gang Yao, Liangrui Peng, Ning Ding

Department of Electronic Engineering, Tsinghua University, Beijing, China

ABSTRACT

In this paper, a multi-modal spatial-temporal representation learning method named VoxelFormer is proposed for 3D object detection in driving scenes. The lidar data are voxelized and Transformer-based network architecture is employed as the backbone to overcome the challenges brought by the sparse and irregular lidar data. Since the quality of image features can be influenced by certain factors, such as occlusion, depth-aware deformable attention is introduced in multi-modal fusion to choose better image features for 3D voxels according to the estimated depth of image pixels and depth of 3D voxels. Multi-modal spatial contextual features at both voxel and region levels are learned by several stages of VoxelFormer block. To get the spatial-temporal representations, previous region-level representations are integrated into current region-level representations. To utilize both voxel and region level features, a coarse-to-fine detection strategy is introduce for more accurate detection. Experiments are conducted on the nuScenes dataset to demonstrate the effectiveness of the proposed method.

Index Terms**—** Multi-modal fusion, spatial-temporal representation learning, 3D object detection

1. INTRODUCTION

Detecting objects in 3D scenes plays an important role in many real-world applications, such as autonomous driving, robot control, etc. Most existing lidar-based 3D object detection methods [1, 2, 3, 4, 5, 6, 7, 8] are based on voxelization and 3D sparse convolution. Transformer [9] has been widely used in natural language processing and computer vision (e.g., image and video). The Transformer-based model is good at capturing large context and fits well into modeling dynamic data, such as the sparse point cloud data. However, it is challenging to apply Transformer on large-scale point cloud data. Recently, some efforts have been made to replace 3D sparse convolution with Transformer. These methods [10, 11, 12] project the point cloud into bird's eye view (BEV) and divide the BEV space into non-overlapping windows with the same spatial sizes. However, projecting the point cloud into BEV will lose some height information and these methods usually utilize large voxel size (0.32m, 0.32m, 6m) which is not effective for small object detection.

Moreover, image features are not used in the above methods. Images can provide rich texture information which is important for accurate and reliable 3D predictions. Early methods such as PointPainting [13] and PointAugmenting [14] directly concatenate raw 3D points with 2D semantics. Some methods [15, 16, 17, 18] project the camera view features to the 3D features using the predicted depths. Then these features are projected to BEV and fused with the lidar BEV map. However, these methods only fuse multi-modal features in the BEV level, losing the benefits of different levels of feature fusion in 3D voxel space.

In this paper, we propose a multi-modal spatial-temporal representation learning method name VoxelFormer for 3D object detection. Different from other transformer-based methods [10, 11, 12], we first divide the lidar points into voxels with smaller voxel size (0.075m, 0.075m, 0.2m) to learn more fine-grained representations. We propose a depth-aware multi-modal deformable attenion to select proper image features when fusion lidar features and multi-view image features. Due to the computational complexity of Transformer, voxels are grouped into regions and self-attention mechanism is used to extract local voxel features. However, the above voxel representations have two issues. First, connections between different regions are not established. Second, only extracting image features for non-empty voxels will lose some information due to the sparsity of lidar data. Therefore, we introduce learnable meta tokens to alleviate these two issues. The voxel features in the same region are aggregated into tokens. Connections between regions are established via Swin Transformer block and the outputs are used to augment the voxel data. To increase the diversity of tokens, tokens are decorated by positional embedding of points which are uniformly sampled within the region. On the one hand, tokens located at different positions can aggregate voxel features from different perspectives. On the other hand, since some sampled points are located in areas without lidar data, performing multi-modal fusion at token level can provide more texture information for representation learning. Since the ego vechicle is moving, features at different time steps can provide diverse information. Therefore, previous information is integrated into current representations for 3D proposal generation. For better detection of small objects, fine-grained voxel-level fea-

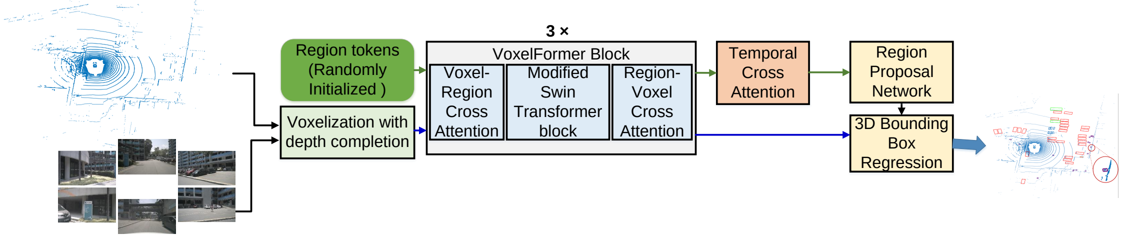

3 ×
Region tokens (Randomly Initialized)
Voxelization with depth completion
VoxelFormer Block
Voxel-Region Cross Attention
Modified Swin Transformer block
Region-Voxel Cross Attention
Temporal Cross Attention
Region Proposal Network
3D Bounding Box Regression

Fig. 1. The architecture of our proposed network.

tures are used for 3D box regression.

To sum up, our main contributions are threefold:

- A novel transformer-based method is proposed to learn multi-modal spatial representations from lidar data and multi-view images by incorporating cross-attention mechanism for voxel level and region level tokens.
- A method to utilize temporal contextual information is proposed by introducing deformable cross attention mechanism for region-level spatial representations at adjacent time steps.
- Experiments on the nuScenes dataset have demonstrated the effectiveness of the proposed method. It can improve detection performance for some small objects.

2. METHOD

2.1. VoxelFormer Backbone

Due to the large computational cost of large-scale lidar data, directly using Transformer to extract features is difficult. For a given lidar point cloud, we first voxelize the points and partition the non-empty voxels into non-overlapping regions. The feature $u \in \mathcal{R}^d$ of each voxel is obtained by a linear layer, where d is the dimension. To learn local voxel features, we perform multi-head self-attention on voxels within the same region. To aggregate local features in each region, learnable meta tokens are introduced. Specifically, given M tokens $T \in \mathcal{R}^{M\times d}$ and N voxel features $U = \{u_1, ..., u_N\} \in \mathcal{R}^{N\times d}$ with 3D coordinates $U_c \in \mathcal{R}^{N\times 3}$ in the same region, M positions $T_c \in \mathcal{R}^{M\times 3}$ are uniformly sampled within the region as the 3D coordinates of tokens to increase the diversity of tokens. The process of voxel feature aggregation can be formulated as:

$$T' = CA(q = T + \phi(T_c), k = v = U + \phi(U_c)), \tag{1}$$

where ϕ is a linear layer to encode positions into d dimensions and CA denotes the cross attention:

$$CA(q, k, v) = q + \text{softmax}(qW_qW_k^Tk^T/\sqrt{d})vW_v, \tag{2}$$

where $W_q, W_k, W_v \in \mathcal{R}^{d\times d}$ are linear projection matrices.

We can generate region-specific tokens for all regions in this way. To establish connections between different regions, a swin transformer [19] network is used. Since previous voxel features only contain local information which is not effective for estimating the whole geometric structures of objects, the updated tokens will be used to augment voxel features. The process of voxel feature augment can be formulated as:

$$U' = CA(q = U + \psi(U_c), k = v = T' + \psi(T_c')), \tag{3}$$

where ψ is a linear layer to encode positions.

2.2. Multi-modal Feature Fusion

Since the lidar points are sparser than image pixels, simply projecting points onto 2D images to collect image features will lose some information. We can use a linear projection layer to generate offset position Δp and collect contextual information according to the offset. However, as shown in Fig. 2, the offset sampled feature is not reliable enough because of some reasons, such as occlusion. If the difference of depth between a position in the image and a 3D voxel is large, the corresponding image feature will have less relation to the voxel features. Therefore, we propose depth-aware deformable attention to extract multi-view image features $\mathcal{I}$ based on the depth map D estimated by CompletionFormer [20]. Given a voxel with feature u and depth u_d, its projected 2D image coordinate is p. 2D Offset positions $\{\Delta p_1, \Delta p_2, ..., \Delta p_k\}$ is generated by a linear layer over voxel feature u. The offset sampled image features $\mathcal{I}(p + \Delta p_i)$ and depth $D(p + \Delta p_i)$ are obtained and then used to estimate the similarity with 3D voxels for $i = 1, ..., k$:

$$S_i = \Phi(\text{Concat}(u, \mathcal{I}(p+\Delta p_i))) + \varphi(u_d - D(p+\Delta p_i)) \tag{4}$$

where Φ and φ are 2-layer MLP to output a single value respectively and k is set to 4 in our work. The softmax function is used to normalize the similarities between k sampling features and the normalized similarities are used to re-weight the sampled image features $\mathcal{I}(p + \Delta p_i)$. The reweighted image features are summed and then concatenated with voxel features to obtain final unified features.

Besides, some tokens introduced in Sec 2.2 are located in areas without lidar data. Therefore, we also perform multi-modal fusion on tokens and images to provide more texture information for representation learning.

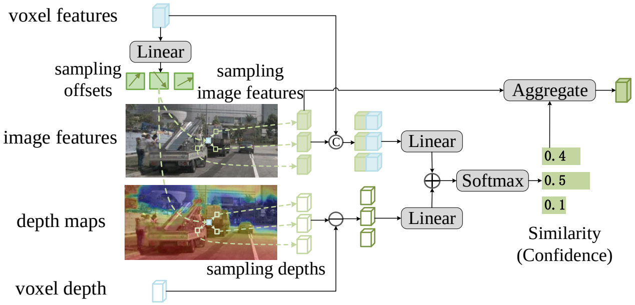


Fig. 2. Illustration of depth-aware multi-modal feature fusion. © denotes concatenation, ⊖ denotes substraction and ⊕ denotes summation.

2.3. Object Detection with Temporal Fusion

For more accurate detection, We introduce a coarse-to-fine object detection head where the region token features and fine-grained voxel features are used to generate 3D proposals and final regression results respectively. In the first stage, the region token features are concatenated along the height direction to obtain a BEV map. Since features at different time steps can provide diverse information, historical BEV features are integrated into current BEV features. Inspired by BEVFormer [21], we introduce deformable attention to better align moving objects at different time steps. Let $X_t, X_{t-1} \in \mathcal{R}^{C \times H \times W}$ denote the BEV feature map at time t and $t-1$, Q_p denotes the query located at $p = (x, y)$ in X_t. According to the motion $(\Delta x, \Delta y)$ of the ego car, its location in X_{t-1} can be calculated as $p' = (x + \Delta x, y + \Delta y)$. The feature at time $t-1$ is calculated by:

$$Q_{p'} = \sum_{l=1}^{L} A_l X_{t-1}(p' + \Delta p_l), \tag{5}$$

where L is the number of sampling keys. Δp_l and A_l denote the sampling offset and attention weight of the k-th sampling key. Both Δp_l and A_l are obtained by a linear layer over the query Q_p. For each pixel in X_t, its corresponding feature in X_{t-1} can be calculated via the above calculations. The previous feature is fused with the current feature by residual connection. The updated BEV map is used to predict a heatmap for classification and top-K queries are selected according to the classification scores. The queries and BEV map are fed to the Transformer decoder to generate coarse predictions.

In the second stage, we can learn more detailed geometric information from the voxel features of each object. We first crop the voxels in all proposal boxes generated by the first stage. Since the number of voxels is different in each proposal, we uniformly sample $n \times n \times n$ grid points G within the 3D proposal and aggregate the voxel features into the grid points by cross attention. Then, we conduct 3-layer self-attention on the grid points to further explore the geometric structures of 3D proposal boxes. The outputs are fed to a 3-layer feed-forward network to obtain final predictions.

2.4. Loss Function

Following TransFusion [3], we compute focal loss [22] for the classification and an L1 loss for the bounding box regression.

3. EXPERIMENT

3.1. Dataset and Metrics

The nuScenes dataset is a large-scale autonomous driving benchmark including 1000 driving scenarios in total, which are split into 700, 150, and 150 scenes for training, validation, and testing, respectively. For detection, mean Average Precision (mAP) and the nuScenes Detection Score (NDS) are usually used as the official metrics.

3.2. Implementation Details

We use Swin Transformer [19] with FPN [25] as the image backbone. The region size N is set to (8, 8, 20) and the number of region tokens M is set to 8. The grid points size n on the second detection stage is set to 4. Our model training has two stages: (1) We first train a single-frame detector for 20 epochs. (2) We then freeze the feature extraction backbone and train the detection head using two consecutive frames of another 10 epochs. Those networks are trained with batch size 1 on 4 NVIDIA V100 GPUs. We do not use Test-Time Augmentation or multi-model ensemble during inference.

3.3. Results on the nuScenes dataset

As shown in Table 1, Our method outperforms state-of-the-art methods on the NuScenes validation set and test set. Especially for the classes with similar sizes, such as vehicle, bus, and trailer, our method achieves great improvement on the NuScenes test set. The Figure 2.4 shows that our method can detect some small objects better.

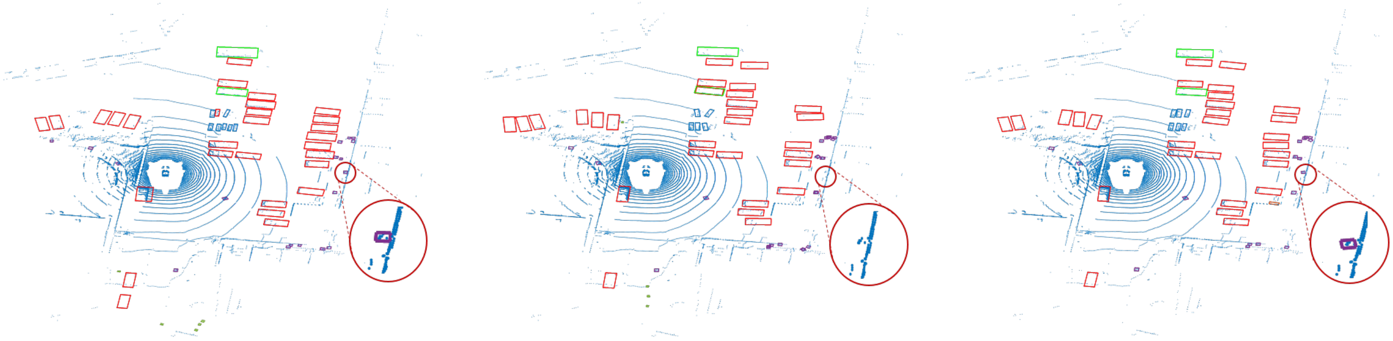
3.4. Ablation Studies

We conduct ablation studies on the nuScenes dataset to pinpoint the improvements. We use the 1/6 training data for efficiency and the whole validation set for testing.

In Table 2, we evaluate the impact of each proposed module on accuracy. The region2voxel cross attention and temporal fusion are removed in the baseline model and it only takes

Table 1. Comparison with state-of-the-art methods on the nuScenes validation (top) and test (bottom) set. Note that these are results without ensemble or test-time augmentation. Construction vehicle (C.V.), pedestrian (Ped.), traffic cone (T.C.).

Method	mAP	NDS	Car	Truck	C.V.	Bus	Trailer	Barrier	Motor.	Bike	Ped.	T.C.
FUTR3D [23]	64.2	68.0	86.3	61.5	26.0	71.9	42.1	64.4	73.6	63.3	82.6	70.1
BEVFusion [15]	67.9	71.0	88.6	65.0	28.1	75.4	41.4	72.2	76.7	65.8	88.7	76.9
Ours	70.1	72.7	89.6	66.3	32.1	78.8	46.9	72.2	78.8	66.0	89.7	80.2
TransFusion [3]	68.9	71.7	87.1	60.0	33.1	68.3	60.8	78.1	73.6	52.9	88.4	86.7
MVP [24]	66.4	70.5	86.8	58.5	26.1	67.4	57.3	74.8	70.0	49.3	89.1	85.0
BEVFusion [15]	69.2	71.8	88.1	**60.9**	34.4	69.3	62.1	78.2	72.2	52.2	**89.2**	85.2
BEVFusion [16]	70.2	72.9	**88.6**	60.1	39.3	69.8	63.8	**80.0**	74.1	51.0	**89.2**	**86.5**
Ours (VoxelFormer)	**70.9**	**73.8**	88.3	58.6	**41.0**	**72.7**	**67.6**	76.7	**74.7**	**54.4**	89.1	86.4



(a) Ground Truth

(b) Baseline

(c) Ours

Fig. 3. The visualization of results on the nuScenes validation set.

point cloud data as input. (B) adds the region2voxel cross attention compared with (A). Since the contextual information in region tokens can be transferred to the voxel features via the region2voxel cross attention, (B) can achieve better performance. The results of (B) and (C) show that the images can provide more information for accurate object detection. The comparison of (C), (D), and (E) illustrates that the depth information and multi-level features can help the multi-modal fusion. The result of (F) demonstrates that the temporal information from previous frames benefits the detection of the current scene. All proposed modules have important effects on the detection performance from the above analysis.

Table 2. Ablation studies of the proposed module on the nuScenes validation dataset.

	mAP	NDS
(A) baseline	50.3	57.6
(B) lidar only with voxel augment	54.2	61.5
(C) single level multi-modal fusion w/o depth	62.6	66.3
(D) single level multi-modal fusion with depth	63.2	66.7
(E) multi-level multi-modal fusion with depth	64.5	67.3
(F) with temporal fusion	**65.7**	**68.5**

Table 3 shows the impact of the number of region tokens in the backbone. The results show that a small amount of region tokens is not enough to represent voxels in the whole region and a large amount of region tokens will bring more noise. From the results, the region token number set to 8 can achieve better detection performance.

Table 3. Effects of the number M of region tokens.

token number	mAP	NDS
1	63.4	66.4
4	63.8	66.9
8	**64.5**	**67.3**
16	64.2	67.1

4. CONCLUSION

In this paper, we propose a multi-modal spatial-temporal representation learning method named VoxelFormer for 3D object detection. We propose a new transformer architecture to learn spatial contextual information of voxels and regions. We propose a multi-level depth-aware multi-modal fusion network to fuse voxel features and multi-view image features. A coarse-to-fine detection head with temporal fusion is used to generate more accurate detection results.

5. REFERENCES

[1] Alex H Lang, Sourabh Vora, Holger Caesar, et al., "Pointpillars: Fast encoders for object detection from point clouds," in *CVPR*, 2019, pp. 12697–12705.

[2] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl, "Center-based 3d object detection and tracking," in *CVPR*, 2021, pp. 11784–11793.

[3] Xuyang Bai, Zeyu Hu, Xinge Zhu, et al., "Transfusion: Robust lidar-camera fusion for 3d object detection with transformers," in *CVPR*, 2022, pp. 1090–1099.

[4] Yukang Chen, Jianhui Liu, Xiangyu Zhang, Xiaojuan Qi, et al., "Voxelnext: Fully sparse voxelnet for 3d object detection and tracking," in *CVPR*, 2023, pp. 21674–21683.

[5] Zeyu Yang, Jiaqi Chen, Zhenwei Miao, et al., "Deepinteraction: 3d object detection via modality interaction," *NeurIPS*, vol. 35, pp. 1992–2005, 2022.

[6] Yanwei Li, Yilun Chen, Xiaojuan Qi, et al., "Unifying voxel-based representation with transformer for 3d object detection," *NeurIPS*, vol. 35, pp. 18442–18455, 2022.

[7] Zhichao Li, Feng Wang, and Naiyan Wang, "Lidar r-cnn: An efficient and universal 3d object detector," in *CVPR*, 2021, pp. 7546–7555.

[8] Ming Liang, Bin Yang, Shenlong Wang, and Raquel Urtasun, "Deep continuous fusion for multi-sensor 3d object detection," in *ECCV*, 2018, pp. 641–656.

[9] Ashish Vaswani, Noam Shazeer, Niki Parmar, et al., "Attention is all you need," *NeurIPS*, vol. 30, 2017.

[10] Lue Fan, Ziqi Pang, Tianyuan Zhang, et al., "Embracing single stride 3d object detector with sparse transformer," in *CVPR*, 2022, pp. 8458–8468.

[11] Pei Sun, Mingxing Tan, Weiyue Wang, et al., "SWFormer: Sparse window transformer for 3d object detection in point clouds," in *ECCV*, 2022, pp. 426–442.

[12] Zhijian Liu, Xinyu Yang, Haotian Tang, et al., "Flatformer: Flattened window attention for efficient point cloud transformer," in *CVPR*, 2023, pp. 1200–1211.

[13] Sourabh Vora, Alex H Lang, Bassam Helou, and Oscar Beijbom, "Pointpainting: Sequential fusion for 3d object detection," in *CVPR*, 2020, pp. 4604–4612.

[14] Chunwei Wang, Chao Ma, Ming Zhu, and Xiaokang Yang, "Pointaugmenting: Cross-modal augmentation for 3d object detection," in *CVPR*, 2021, pp. 11794–11803.

[15] Tingting Liang, Hongwei Xie, Kaicheng Yu, et al., "Bevfusion: A simple and robust lidar-camera fusion framework," *NeurIPS*, vol. 35, pp. 10421–10434, 2022.

[16] Zhijian Liu, Haotian Tang, Alexander Amini, et al., "Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation," in *ICRA*, 2023, pp. 2774–2781.

[17] Xin Li, Botian Shi, Yuenan Hou, et al., "Homogeneous multi-modal feature fusion and interaction for 3d object detection," in *ECCV*, 2022, pp. 691–707.

[18] Jonah Philion and Sanja Fidler, "Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d," in *ECCV*, 2020, pp. 194–210.

[19] Ze Liu, Yutong Lin, Yue Cao, et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *ICCV*, 2021, pp. 10012–10022.

[20] Youmin Zhang, Xianda Guo, Matteo Poggi, et al., "Completionformer: Depth completion with convolutions and vision transformers," in *CVPR*, 2023, pp. 18527–18536.

[21] Zhiqi Li, Wenhai Wang, Hongyang Li, et al., "Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers," in *ECCV*, 2022, pp. 1–18.

[22] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár, "Focal loss for dense object detection," in *ICCV*, 2017, pp. 2980–2988.

[23] Xuanyao Chen, Tianyuan Zhang, Yue Wang, Yilun Wang, et al., "Futr3d: A unified sensor fusion framework for 3d detection," in *CVPR*, 2023, pp. 172–181.

[24] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl, "Multimodal virtual point 3d detection," *NeurIPS*, vol. 34, pp. 16494–16507, 2021.

[25] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, et al., "Feature pyramid networks for object detection," in *CVPR*, 2017, pp. 2117–2125.