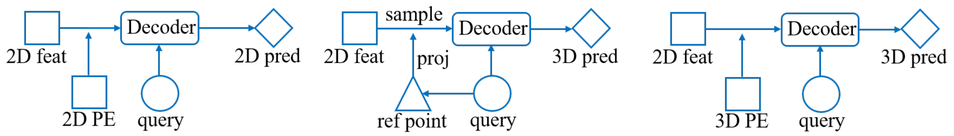
PETR: Position Embedding Transformation for Multi-View 3D Object Detection

Yingfei Liu, Tiancai Wang, Xiangyu Zhang, Jian Sun

MEGVII Technology

Problem Definition and Contribution

Goal: Perform 3D object detection from multi-view images like 2D detectors.

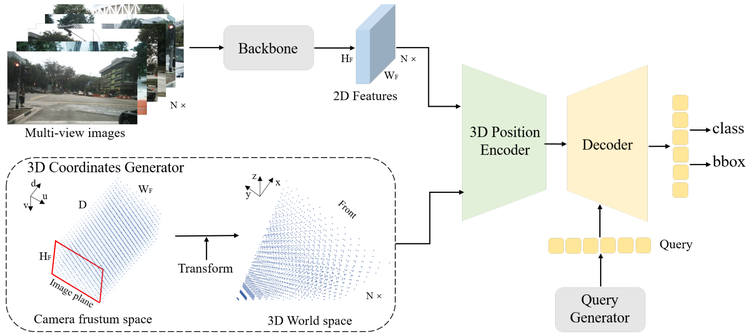


Key Contributions: A simple and elegant framework for vison 3D object detection that

- without the online 2D-to-3D transformation and feature sampling.
- introduce 3D position embedding for multi-view 3D object detection. The object quries can interact with 3D position-aware features directly.
- achieve state-of-the-art performance (50.4%NDS and 44.1% mAP) on standard nuScenes dataset and ranks 1st place on 3D object detection leaderboard.

Method

Network Architecture: PETR keeps the end-to-end spirit of original DETR while avoiding the complex 2D-to-3D projection and feature sampling.

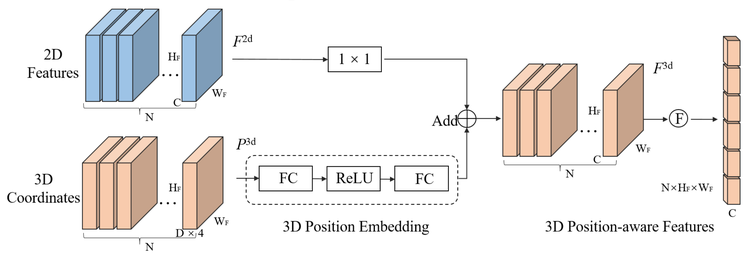


3D Coordinates Generator: Each point in the meshgrid can be represented as $p_j^m = (u_j \times d_j, v_j \times d_j, d_j, 1)^T$, where (u_j, v_j) is a pixel coordinate in the image, d_j is the depth value. The corresponding 3D coordinate $p_{i,j}^{3d} = (x_{i,j}, y_{i,j}, z_{i,j}, 1)^T$ can be calculated as:

$$p_{i,j}^{3d} = K_i^{-1} p_j^m \tag{1}$$

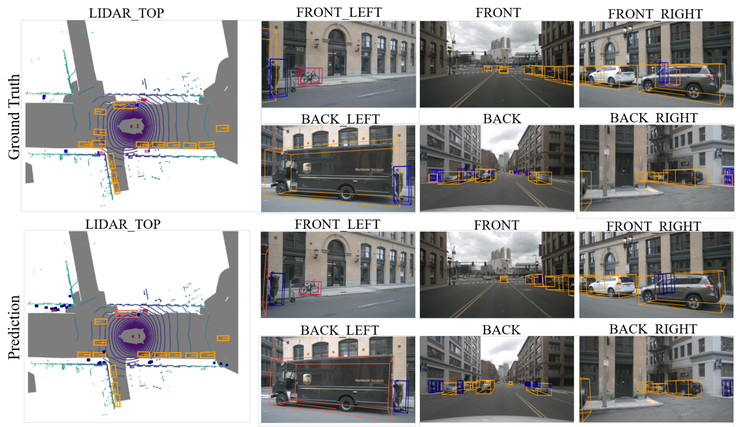
where $K_i \in R^{4\times4}$ is the transformation matrix of i-th view.

3D Position Encoder: 3D Position Encoder is a simple MLP network. The 3D position embeddings are added with the 2D image features.

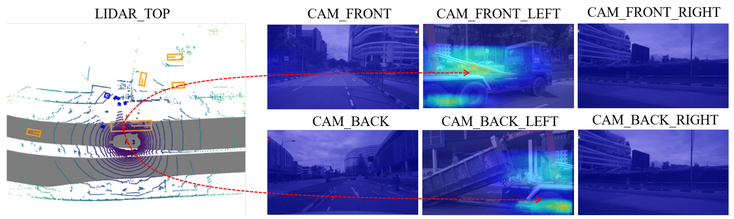


Experiments & Results

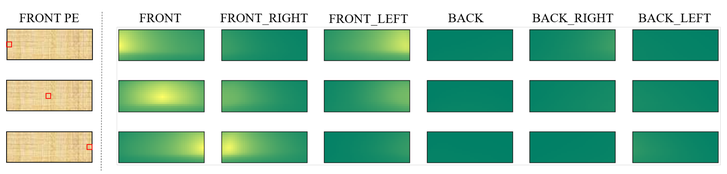
Qualitative analysis of detection results in BEV and image views:



Visualization of attention maps:

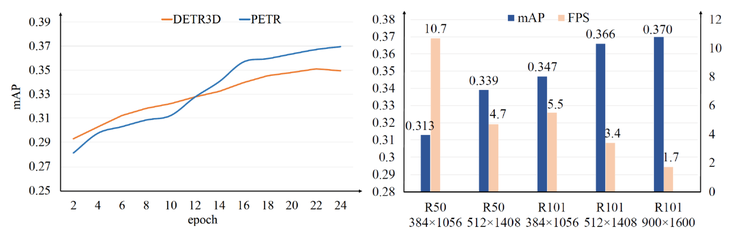


3D position embedding similarity:



The red points are selected positions in the front view. We calculated the similarity between the position embedding of these selected positions and all image views.

Convergence and speed analysis on PETR:



Comparison of recent works on the nuScenes test set:

Methods	Backbone	NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
FCOS3D‡	Res-101	0.428	0.358	0.690	0.249	0.452	1.434	**0.124**
PGD‡	Res-101	0.448	0.386	0.626	0.245	0.451	1.509	0.127
DD3D∗‡	V2-99	0.477	0.418	0.572	0.249	**0.368**	1.014	**0.124**
DETR3D∗	V2-99	0.479	0.412	0.641	0.255	0.394	0.845	0.133
BEVDet	Swin-S	0.463	0.398	0.556	**0.239**	0.414	1.010	0.153
BEVDet∗	V2-99	0.488	0.424	**0.524**	0.242	0.373	0.950	0.148
PETR	Res-101	0.455	0.391	0.647	0.251	0.433	0.933	0.143
PETR	Swin-T	0.450	0.411	0.664	0.256	0.522	0.971	0.137
PETR	Swin-S	0.481	0.434	0.641	0.248	0.437	0.894	0.143
PETR	Swin-B	0.483	**0.445**	0.627	0.249	0.449	0.927	0.141
PETR∗	V2-99	**0.504**	0.441	0.593	0.249	0.383	**0.808**	0.132

The impact of 3D PE:

PE	2D	MV	3D	NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
1	✓			0.208	0.069	1.165	0.290	0.773	0.936	0.259
2	✓	✓		0.224	0.089	1.165	0.287	0.738	**0.929**	0.251
3			✓	0.356	0.305	**0.835**	**0.238**	0.639	0.971	**0.237**
4	✓		✓	0.351	0.305	0.838	0.283	**0.633**	1.048	0.256
5	✓	✓	✓	**0.359**	**0.309**	0.844	0.278	0.653	0.945	0.241

The ablation studies:

PE Networks	NDS↑	mAP↑	mATE↓
None	0.311	0.256	1.00
1×1 ReLU 1×1	**0.359**	**0.309**	**0.839**
3×3 ReLU 3×3	0.017	0.000	1.054

(a) The network to generate the 3D PE. "None" means that the normalized 3D coordinates are directly used as 3D PE.

Fusion Ways	NDS↑	mAP↑	mATE↓
Add	**0.359**	**0.309**	0.839
Concat	0.358	0.309	**0.832**
Multiply	0.357	0.303	0.848

(b) Different ways to fuse the 2D multi-view features with 3D PE in the 3D position encoder.

Anc-Points	NDS↑	mAP↑	mATE↓
None	-	-	-
Fix-BEV	0.337	0.295	0.852
Fix-3D	0.343	0.303	0.864
Learned-3D	**0.359**	**0.309**	**0.839**

(c) "None" means no anchor points following DETR. "Fix-BEV" and "Fix-3D" mean the grid anchor points in BEV space and 3D space respectively.

Points-Num	NDS↑	mAP↑	mATE↓
600	0.339	0.300	0.847
900	0.351	0.303	0.860
1200	0.354	0.303	0.854
1500	**0.359**	**0.309**	**0.839**

(d) Results with different numbers of anchor points. We explored anchor point numbers ranging from 600 to 1500. More points perform better.

Code & Model: