Learning Ego 3D Representation as Ray Tracing

Jiachen Lu[1], Zheyuan Zhou[1], Xiatian Zhu[2], Hang Xu[3], Li Zhang[1]

[1]Fudan University [2]University of Surrey [3]Huawei Noah's Ark Lab

Limitations of existing dense 3D representation learning strategies

Existing perception methods often rely on error-prone depth estimation of the whole scene or learning sparse virtual 3D representations without the target geometry structure, both of which remain limited in performance and/or capability.

1. The ***first*** strategy relies on pixel-level depth estimation. A downside of these methods is that depth estimation in unconstrained scenes is typically error-prone, which would be further propagated down to the subsequent components.
2. The ***second*** strategy eliminates the depth dimension via directly learning 3D representations from 2D images through architecture innovation. However, their 3D representation is structurally inconsistent with 2D counterparts as no rigorous intrinsic and extrinsic projection can be leveraged.

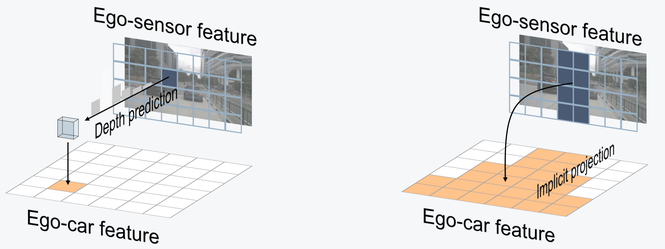
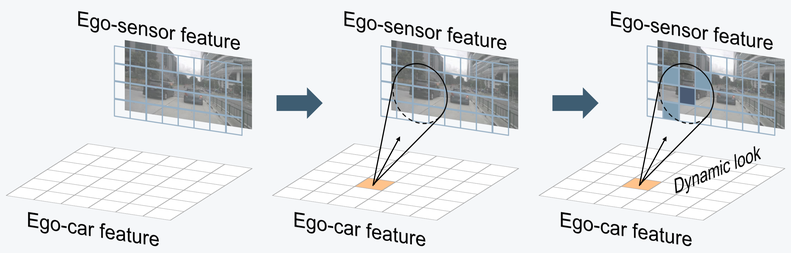


Figure1: Comparison of dense 3D representation learning strategies. (a) The first strategy, is based on dense pixel-level depth estimation. (b) The second strategy represented bypasses the depth estimation by learning implicit 2D-3D projection.

Learning Ego 3D Representation as Ray Tracing



1. We start with introducing a polarized grid of dense "imaginary eyes" for BEV representation, with each eye naturally occupying a specific geometry location with the depth information involved.
2. For learning 3D representation including height information intrinsically absent in BEV, we initialize each eye using a uniform value and leave the eyes to look backward surrounding 2D visual representations subject to the intrinsic and extrinsic 3D-to-2D projection.
3. With the adaptive attention mechanism, eyes focus dynamically on 2D representations and directly learn to approximate missing height information in a data driven manner.

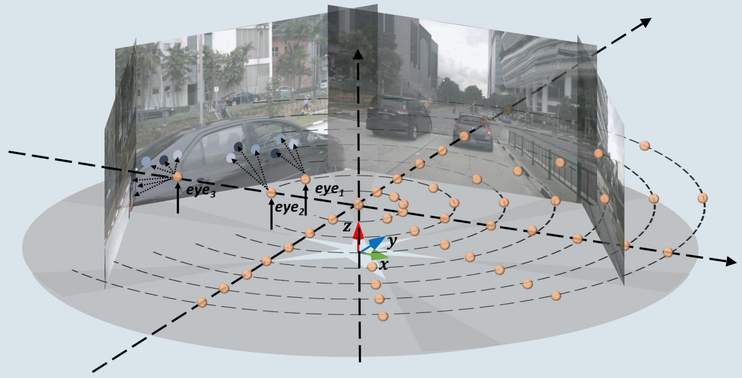


Figure2: An illustration of tracing 3D backwards to 2D mechanism for imaginary eyes. The golden balls represents the polarized grid of dense ``imaginary eyes". Specially, for eyes have multiple visible images, they backtrack to multiple images, while eyes having only single visible image backtrack to single image.

Multi-view multi-scale adaptive attention (MVAA)

MVAA is the core of transferring 2D representation into 3D. We formulate the learning of these imaginary eyes in an adaptive self-attention detection framework. The process can be expressed as:

$$\text{MVAA}\left(\mathbf{y}_q, \mathbf{r}_q, \left\{\left\{x_l^{(t)}\right\}_{l=1}^{N_{\text{scale}}}\right\}_{t=1}^{N_{\text{view}}}\right) = \text{concat}_{h\in\{N_h\}}\, \mathbf{W}_h \left[\sum_{t\in\{N_{\text{scale}}\}} \sum_{t\in I_q} \sum_{k\in\{N_{\text{point}}\}} \mathbf{A}_{hltk} \cdot \mathbf{W'}_h \phi\left(x_l^{(t)}, \mathbf{M}^{(t)}\mathbf{r} + \Delta\mathbf{r}_{hvlk}\right)\right]$$

$\mathbf{y} \in \mathbb{R}^{C\times N_{eye}}$ is the eye queries, $\mathrm{r} \in \mathbb{R}^{3\times N_{eye}}$ is the location of eyes in ego car coordinate. Formally, each eye(i.e., query) will dynamically choose N_{point} feature points at N_{scale} scales of 2D image representation. $\mathbf{A}$ and $\mathbf{\Delta r}$ are based on learnable parameters:

$$\mathbf{A} = \text{softmax}_{tk}\left(\mathbf{W}_q^{(A)}\mathbf{y}_q\right), \qquad \Delta r = \mathbf{W}_q^{(\mathrm{r})}\mathbf{y}_q + \mathbf{b}_q^{(\mathrm{r})}$$

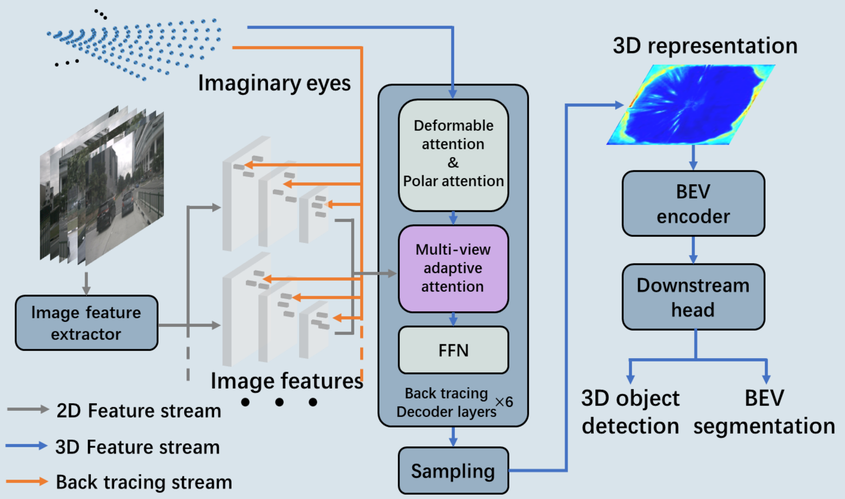


Figure3: Our pipeline comprises two stages: learning ego 3D representation from 2D features and executing multiple downstream tasks based on 3D representation. The gray lines represent the 2D feature stream while the blue lines represent the 3D feature stream. Besides, the orange lines specify our back tracing path.

3D detection results on the nuScenes validation set

Methods	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓	mAP↑	NDS↑
FCOS3D [29]	0.790	**0.261**	0.499	1.286	**0.167**	0.298	0.377
DETR3D [32]	0.860	0.278	**0.327**	**0.967**	0.235	0.303	0.374
PGD [28]	0.732	0.263	0.423	1.285	0.172	0.336	0.409
Ego3RT(Ours)	**0.714**	0.275	0.421	0.988	0.292	**0.355**	**0.409**
FCOS3D† [29]	0.754	**0.260**	0.486	1.331	**0.158**	0.321	0.395
DETR3D† [32]	0.765	0.267	0.392	0.876	0.211	0.347	0.422
PGD† [28]	0.667	0.264	0.435	1.276	0.177	0.358	0.425
Ego3RT(Ours)†	**0.657**	0.268	**0.391**	**0.850**	0.206	**0.375**	**0.450**
Ego3RT(Ours)‡	**0.582**	0.272	**0.316**	**0.683**	0.202	**0.478**	**0.534**

3D detection results on the nuScenes validation set

Method	multi?	Drivable	Crossing	Walkway	Carpark	Divider
VED [15]	✗	54.7	12.0	20.7	13.5	-
VPN [18]	✗	58.0	27.3	29.4	12.3	-
PON [23]	✗	60.4	28.0	31.0	18.4	-
OFT [24]	✗	62.4	30.9	34.5	23.5	-
LSF [7]	✗	61.1	33.5	37.8	25.4	-
Image2Map [25]	✗	74.5	**36.6**	35.9	31.3	-
OFT [24]	✓	71.7	-	-	-	18.0
LSS [20]	✓	72.9	-	-	-	20.0
Ego3RT(Ours)	✓	**79.6**	**48.3**	**52.0**	**50.3**	**47.5**
Ego3RT(Ours) ¶	✓	**74.6**	33.0	**42.6**	**44.1**	**36.6**

Qualitative results on nuScenes dataset

Project Page: https://fudan-zvg.github.io/Ego3RT/