VisionPAD: A Vision-Centric Pre-training Paradigm for Autonomous Driving

Haiming Zhang [1,2], *Wending Zhou* [1,2], *Yiyao Zhu* [3], *Xu Yan* [4] †, *Jiantao Gao* [4], *Dongfeng Bai* [4], *Yingjie Cai* [4], *Bingbing Liu* [4], *Shuguang Cui* [2,1], ***Zhen Li*** [2,1]†

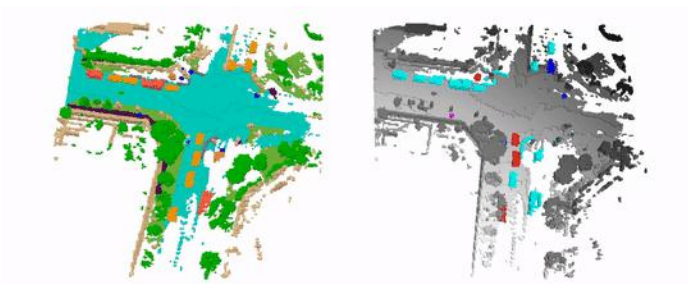
[1] The Future Network of Intelligence Institute, The Chinese University of Hong Kong (Shenzhen),
[2] School of Science and Engineering, The Chinese University of Hong Kong (Shenzhen),
[3] HKUST
[4] Huawei Noah's Ark Lab

Background

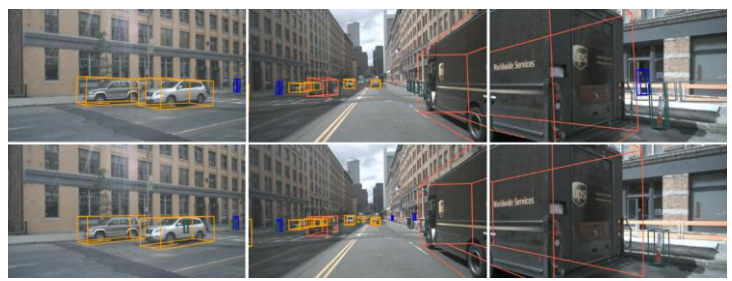
Vision-centric 3D Perception Tasks:

- Inputs: Multi-view camera images
- Outputs: 3D bounding boxes (3D object detection), 3D semantic occupancy, map segmentation
- Advantages: cost-effectiveness, general object representation, suitable for unified models

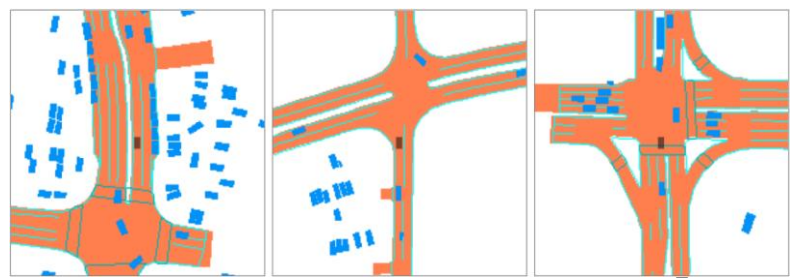
Multi-view images



3D semantic occupancy prediction



3D object detection



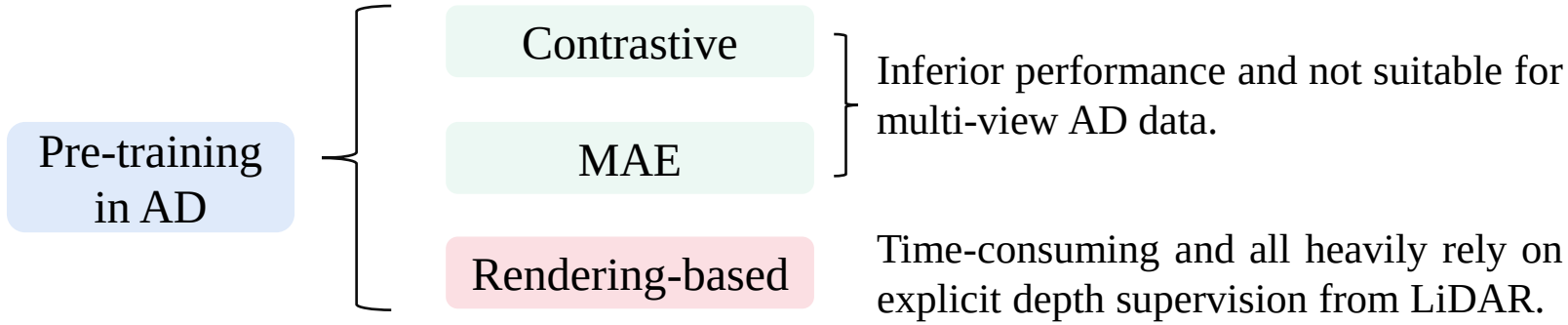
Map segmenation

Challenges

Challenges of Scaling up Vision-centric Perception Models:

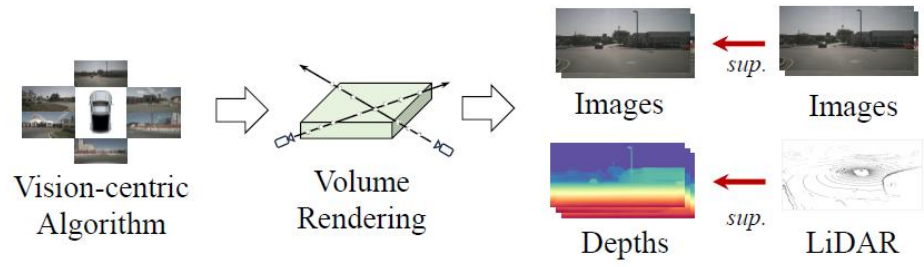
- Lack of large-scale 3D annotations;
- Time-consuming and high-load to train models with large parameters;

How to solve? Pre-training is an effective method to rescue.

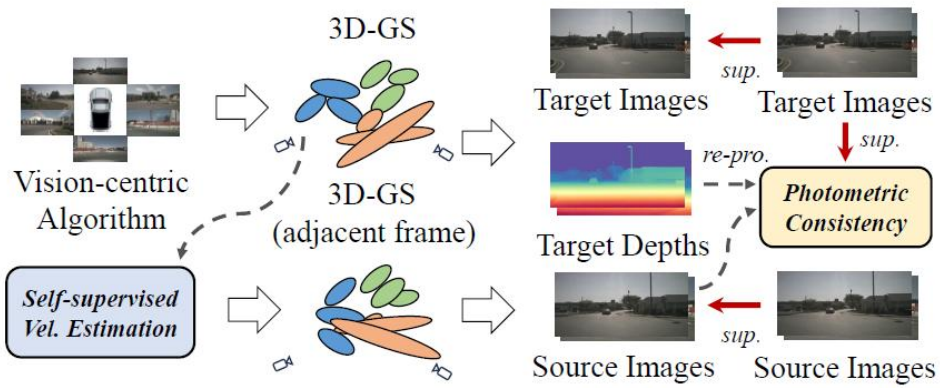


Motivation

Could we design an efficient self-supervised pre-training paradigm solely rely on vision inputs?



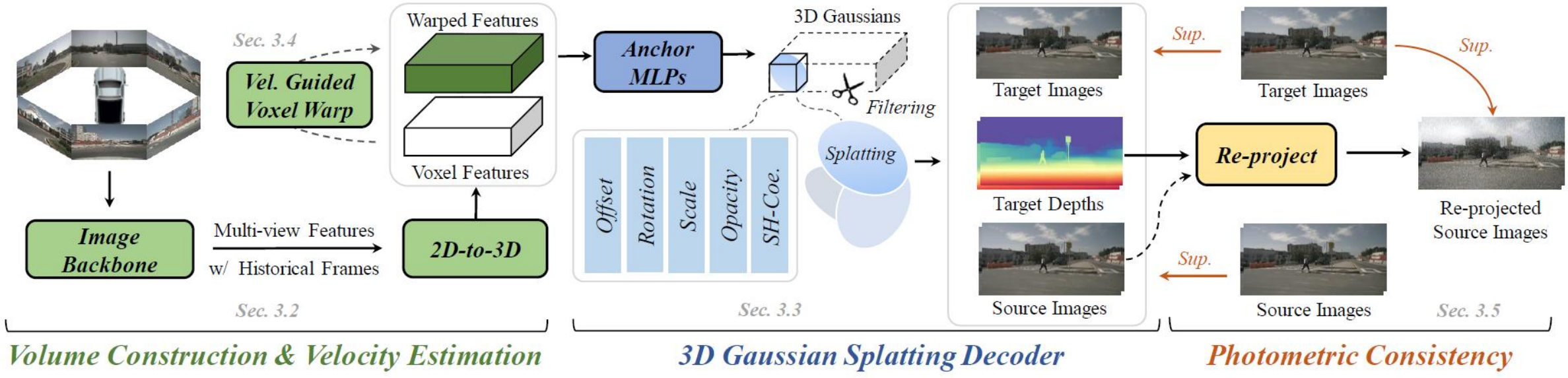
(a) UniPAD: volume rendering with explicit depth supervision



(b) VisionPAD: 3D-GS with sorely vision-centric supervision

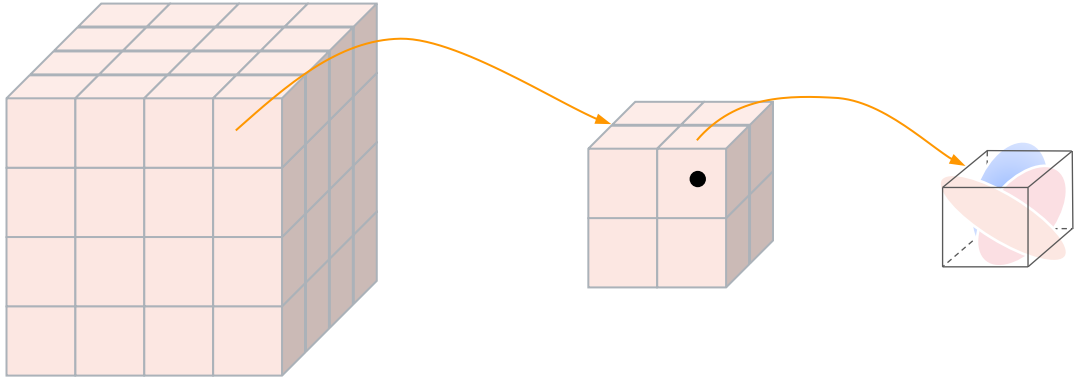
- First introduce a more efficient **anchor-based 3DGS representation** in vision-centric perception models ;
- Propose a **photometric consistency** module to impose geometric information into the volume feature **without utilizing the LiDAR sensors**;
- The **self-supervised volume velocity estimation module** further enhance the motion cues.

Framework



- Any vision-centric perception models build the volumetric features;
- An efficient anchor-based 3DGS representation built upon the volume feature by shallow MLPs;
- Photometric consistency loss and self-supervised velocity estimation modules ensure the pre-training performances;

3D Gaussian Splatting Representation



$\Delta\mu$ center offset
s scale
r rotation
α opacity
c color

Volume features Voxels as Anchors Anchor-based 3DGS

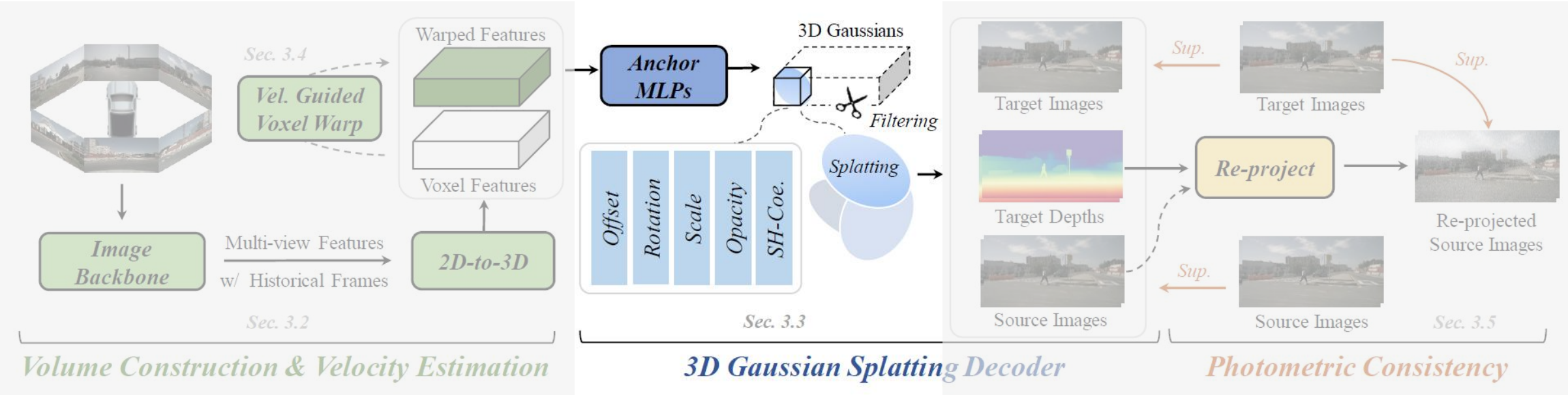
RGB rendering:

$$\mathbf{C}(p) = \sum_{i \in K} c_i \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_i),$$

Depth rendering:

$$\mathbf{D}(p) = \sum_{i \in K} d_i \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j),$$

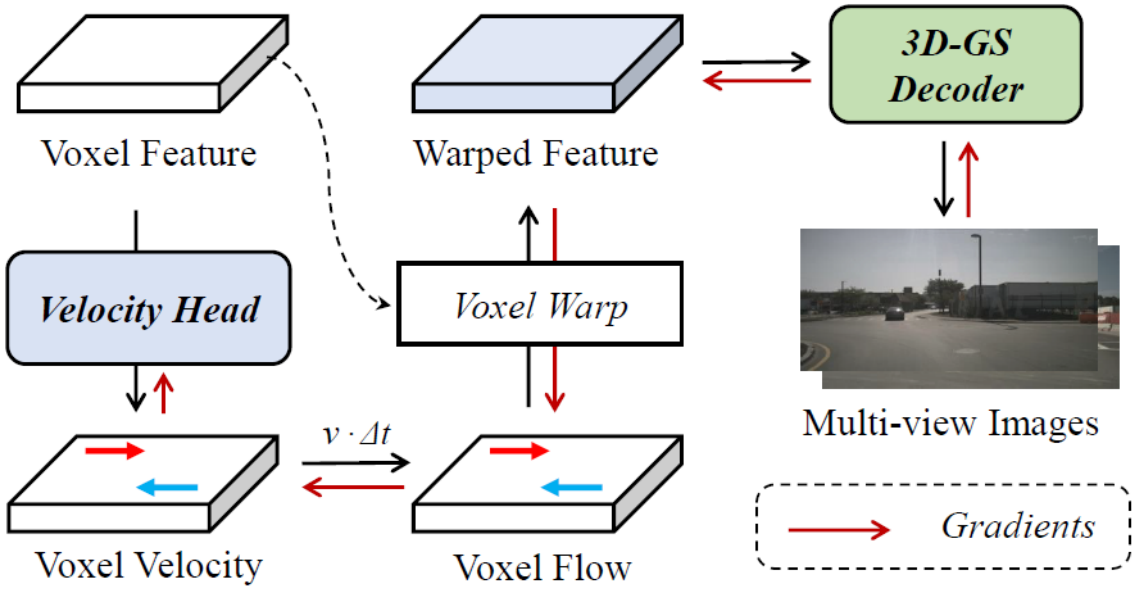
3D Gaussian Splatting Decoder



- Each voxel center severs as an anchor point from which the attributes of multiple Gaussian primitives are predicted;
- The MLPs are utilized to predict these 3DGS attributes;
- To reduce computational overhead during pre-training, we filter low-confidence Gaussians based on their predicted opacity after a tanh activation function that less than 0.

Self-supervised Voxel Velocity Estimation

Purposes: Enriches the volume representation and facilitates understanding of the dynamic scene.



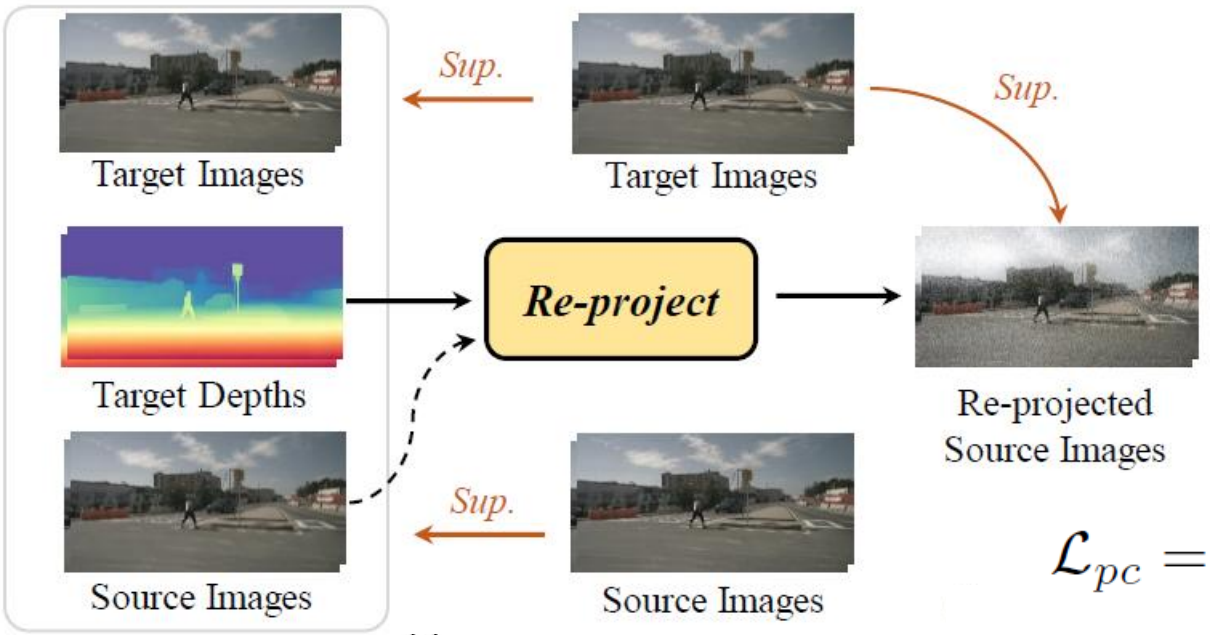
The warped features are utilized in 3DGS decoder with adjacent frame RGB supervision.

Algorithm 1 The pseudocode of voxel velocity estimation.

Input: $\mathcal{V}_t, M$
Output: $\mathcal{V}_{t+n}$

 # predict the absolute flow for each voxel
$\mathcal{F}_t \leftarrow$ **flow_decoder**$(\mathcal{V}_t)$
 # transform the flow into absolute grid displacement to the future
$\hat{\mathcal{F}}_{t+n} \leftarrow \Delta t \cdot n \cdot \mathcal{F}_f$
 # transform the absolute displacement to the future ego coordinate by using the current to future transformation
$\tilde{\mathcal{F}}_{t+n} \leftarrow M \cdot \hat{\mathcal{F}}_{t+n}$
 # warping the current frame volume feature to the future
$\mathcal{V}_{t+n} \leftarrow$ **grid_sample**$(\mathcal{V}_t, \tilde{\mathcal{F}}_{t+n})$
return $\mathcal{V}_{t+n}$

Photometric Consistency



Without rely on any depth ground truth from LiDAR sensors, we project the rendered depth into adjacent **multiple views** and re-sampling the RGB value for self-supervision.

$$\mathbf{I}_{t'\to t} = \mathbf{I}_{t'} \left\langle \text{proj}(\mathbf{D}_t, \mathbf{T}_{t\to t'}, \mathbf{K}) \right\rangle ,$$

$$\mathcal{L}_{pc} = \alpha(1-\text{SSIM}(\mathbf{I}_t, \mathbf{I}_{t'\to t}))+(1-\alpha)\|\mathbf{I}_t-\mathbf{I}_{t'\to t}\|_1,$$

$$\mathbf{D}_t = \text{3DGS}(\mathbf{V}_t, \mathbf{K}_t, \mathbf{T}_t),$$

where $\mathbf{V}_t$, $\mathbf{K}_t$ and $\mathbf{T}_t$ are volume features, camera intrinsics and extrinsics, respectively.

Pre-training Loss

$$\mathcal{L} = \omega_1 \mathcal{L}_{img} + \omega_2 \mathcal{L}_{vel} + \omega_3 \mathcal{L}_{pc},$$

Current frame RGB reconstruction

L1 loss for reconstructed RGB images from warped voxel features

Photometric consistency loss

where ω_1, ω_2 and ω_3 are 0.5, 1, 1, respectively.

Experiments

Datasets and Tasks

- nuScenes: 3D object detection, map segmentation
- nuScenes-Occ3D: 3D semantic occupancy prediction

Metrics

- **3D object detection**: nuScenes Detection Score (NDS) and mean Average Precision (mAP)
- **3D occupancy prediction**: mean Intersection-over-Union (mIoU);
- **Map segmentation**: Intersection-over-Union (IoU)

Finetune Settings

- We strictly follow the official training configurations during fine-tuning without any modifications.

Results

3D object detection results on nuScenes validation set

Methods	Venue	Pre-train Modal	CS	CBGS	NDS (%) ↑	mAP (%) ↑
BEVFormer-S [17]	ECCV'22	-		✓	44.8	37.5
SpatialDETR [7]	ECCV'22	-			42.5	35.1
PETR [20]	ECCV'22	-		✓	44.2	37.0
Ego3RT [22]	ECCV'22	-			45.0	37.5
3DPPE [30]	ICCV'23	-		✓	45.8	39.1
BEVFormerV2 [40]	CVPR'23	-			46.7	39.6
CMT-C [37]	ICCV'23	-		✓	46.0	40.6
FCOS3D	ICCVW'21	-			38.4	31.1
UVTR [14]	NeurIPS'22	-			45.0	37.2
UVTR+UniPAD	CVPR'24	C			44.8 ↓0.2	38.5 ↑1.3
UVTR+VisionPAD (Ours)	-	C			**46.7** ↑1.7	**41.0** ↑3.8
UVTR [14]	NeurIPS'22	-	✓		48.8	39.2
UVTR+UniPAD	CVPR'24	C	✓		48.6 ↓0.2	40.5 ↑0.7
UVTR+UniPAD	CVPR'24	C+L	✓		50.2 ↑1.4	42.8 ↑3.6
UVTR+VisionPAD (Ours)	-	C	✓		49.7 ↑0.9	41.2 ↑2.0
UVTR+VisionPAD (Ours)	-	C+L	✓		**50.4** ↑1.6	**43.1** ↑3.9

Table 1. **3D object detection performance on the nuScenes *val* set.** We benchmark against state-of-the-art methods across various modalities *without* test-time augmentation. Our approach achieves superior performance among existing vision-centric methods. "CS" denotes models trained with camera sweeps (two historical frames) as input. "CBGS" refers to the class-balanced grouping and sampling [47].

Results

3D semantic occupancy prediction results on nuScenes-Occ3D validation set

Methods	Venue	Backbone	Image Size	Pre-train by Det.	mIoU (%) ↑
BEVFormer [17]	ECCV'22	R101	900 × 1600	✓	39.3
TPVFormer [11]	CVPR'23	R50	900 × 1600	✓	34.2
FB-Occ (16f) [18]	CVPRW'23	R50	384 × 704	✓	39.1
RenderOcc [27]	ICRA'24	Swin-B	512 × 1408	✓	24.5
SparseOcc (16f) [19]	ECCV'24	R50	256 × 704	-	30.6
OPUS (8f) [34]	NeurIPS'24	R50	256 × 704	-	36.2
UVTR† [14]	NeurIPS'22	ConvNeXt-S	900 × 1600	-	30.1
UVTR+UniPAD† [41]	CVPR'24	ConvNeXt-S	900 × 1600	-	31.0 ↑0.9
UVTR+VisionPAD (Ours)	-	ConvNeXt-S	900 × 1600	-	**35.4** ↑5.4
BEVDet-Occ (8f) [10]	ArXiv'22	R50	384 × 704	✓	39.3
BEVDet-Occ (8f)+VisionPAD (Ours)	-	R50	384 × 704	✓	**42.0** ↑2.7

Table 2. **Semantic Occupancy prediction performance on Occ3D *val* set.** "8f" and "16f" indicate models incorporating temporal information from 8 and 16 frames, respectively. Results are largely sourced from OPUS [34], with those marked † indicating our own implementation. "Pre-train by Det." refers to methods initializing with weights pre-trained for 3D object detection.

Results

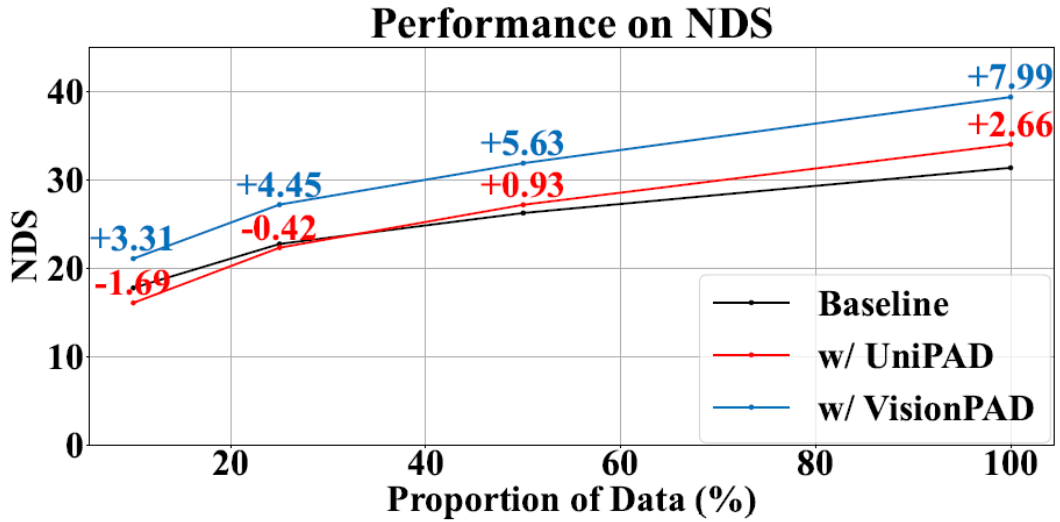
Map segmentation results on nuScenes validation set

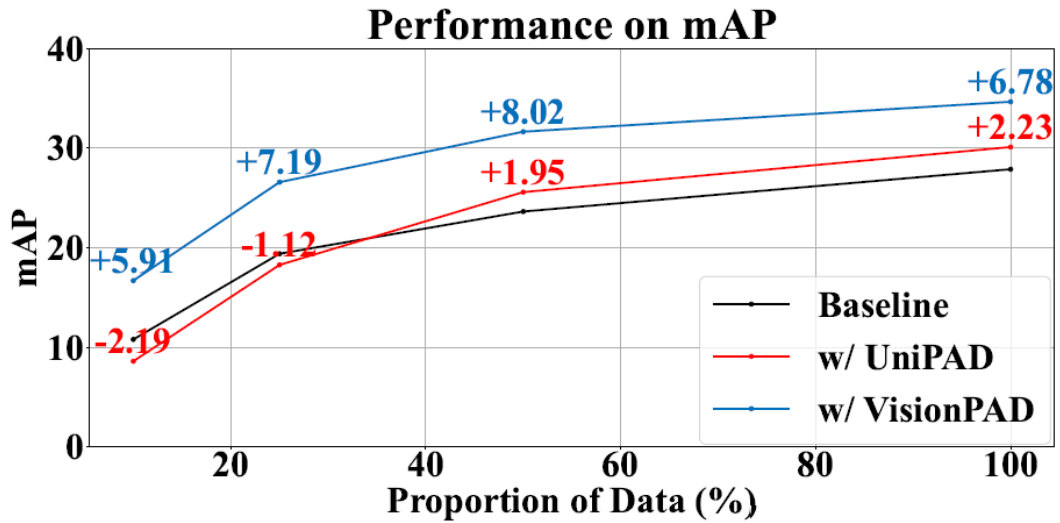
Methods	Backbone	Lanes (%) ↑
UVTR [14]	ConvNeXt-S	15.0
UVTR+UniPAD [41]	ConvNeXt-S	16.3 ↑1.3
UVTR+Ours	ConvNeXt-S	**20.4** ↑5.4

Table 3. **Map segmentation performance.** All reported results utilize the map decoder from UniAD [9], with BEV segmentation lane Intersection over Union (IoU) as the evaluation metric.

Results

Data efficiency analysis with limited data





Ablation

Ablation Study on main designs

Methods	Vol. Rend.	3DGS Dec.	Gaus. Filter	V.V. Est.	P.C.	NDS	mAP
UVTR (baseline) [14]						22.8	19.4
UniPAD [41]	✓					22.3 ↓0.5	18.3 ↓1.1
Model A		✓				22.8 ↑0.0	18.2 ↓1.2
Model B		✓	✓			23.4 ↑0.6	18.9 ↓0.5
Model C		✓	✓	✓		23.6 ↑0.8	20.1 ↑0.7
Model D		✓	✓		✓	26.0 ↑3.2	24.5 ↑5.1
VisionPAD (Ours)		✓	✓	✓	✓	**27.3** ↑4.5	**26.5** ↑7.1

Table 4. **Ablation studies.** We report the NDS and mAP metrics in the nuScenes *val* set for the 3D object detection task. "Dec.", "V.V. Est" and "P.C." denote decoder, voxel velocity estimation and photometric consistency, respectively.

Ablation

Primitives	Variants	Value	NDS	mAP
Mean	Absolute	-	26.8	25.9
	Offset	0.25	27.3	26.5
		0.50	27.2	26.5
Scale	Fixed	0.4	26.9	25.3
	Learnable	[0.1, 0.5]	27.3	26.5
		[0.2, 0.8]	26.8	26.1
Rotation	Fixed	[1, 0, 0, 0]	27.2	26.7
	Learnable	-	27.3	26.5

Ablation Study on various learning objectives for predicting Gaussian primitives

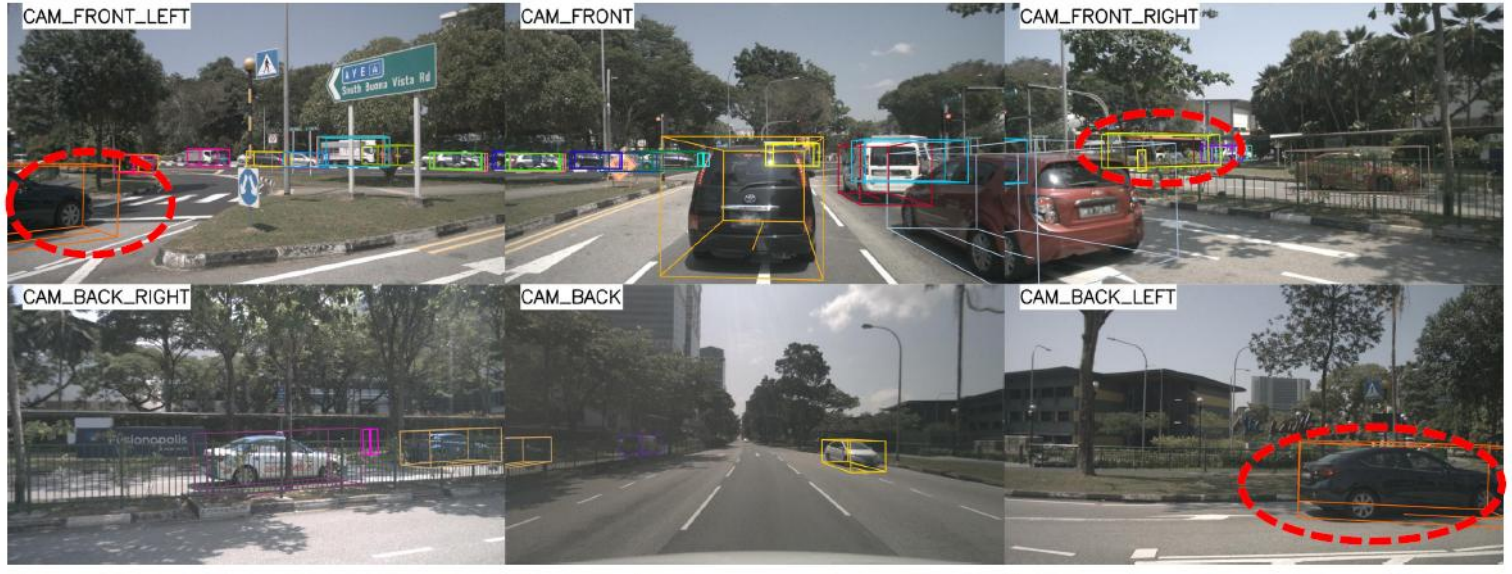
Methods	Decoder	Memory	Latency
UniPAD-C	NeRF	1973MB	900ms
VisionPAD	3D-GS	134MB	70ms

Speed analysis of our method with the UniPAD.

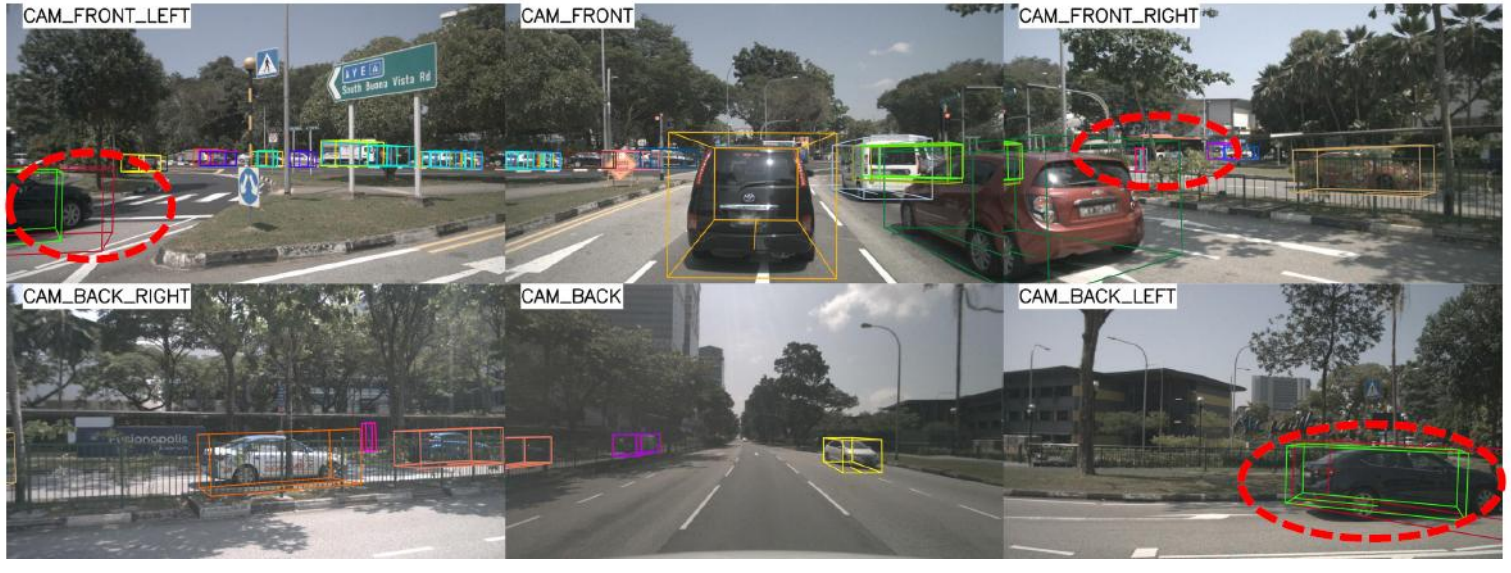
Visualization

CVPR Nashville JUNE 11-15, 2025

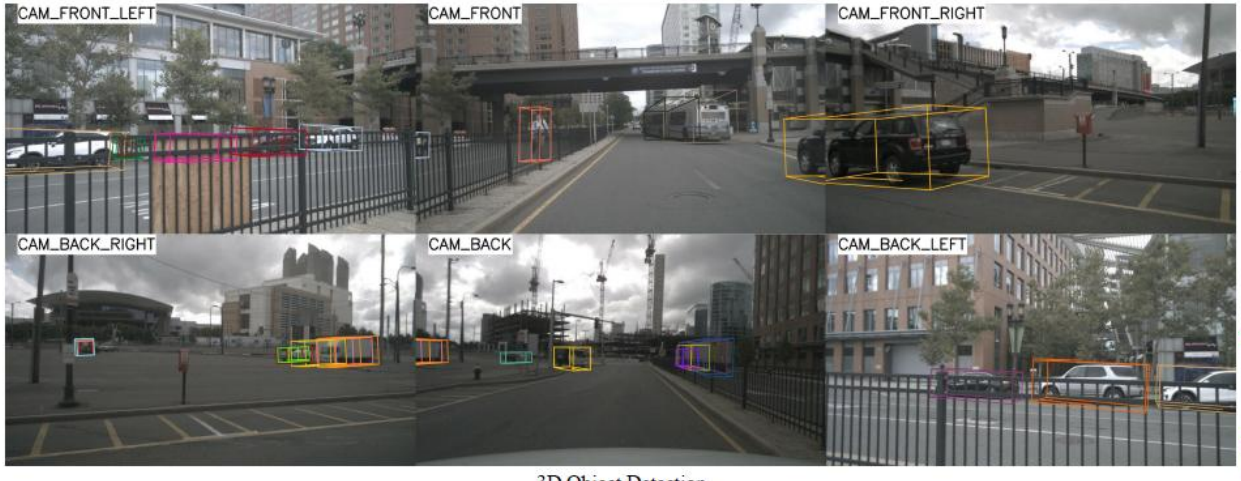
UniPAD



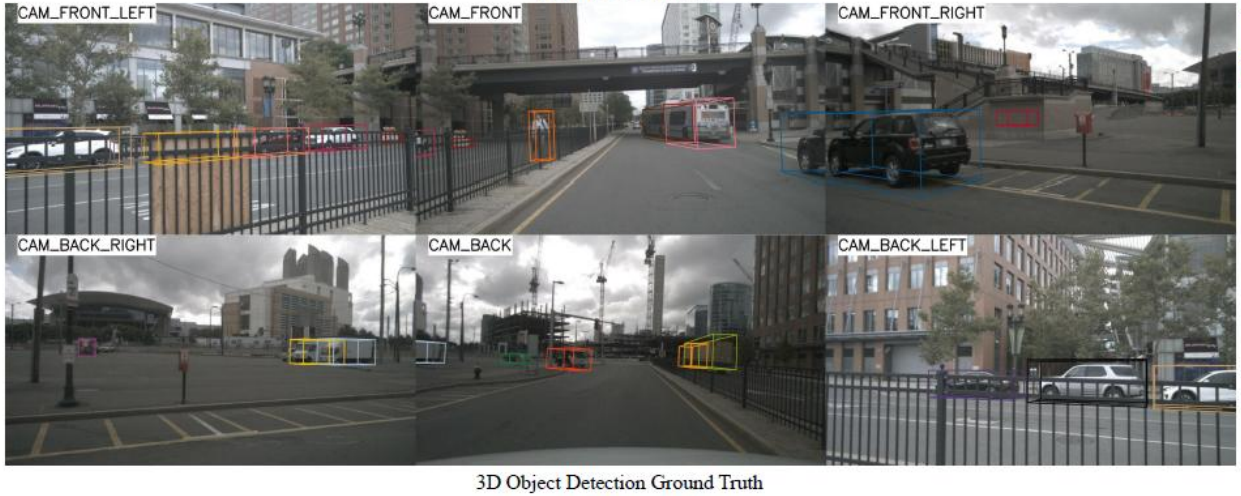
Ours



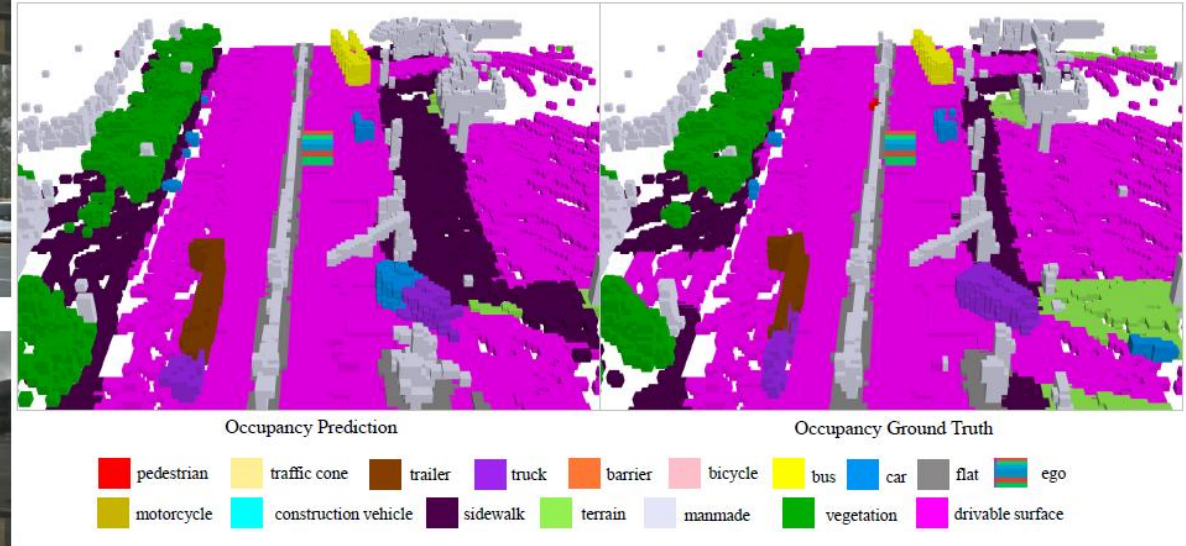
Visualization



3D Object Detection



3D Object Detection Ground Truth

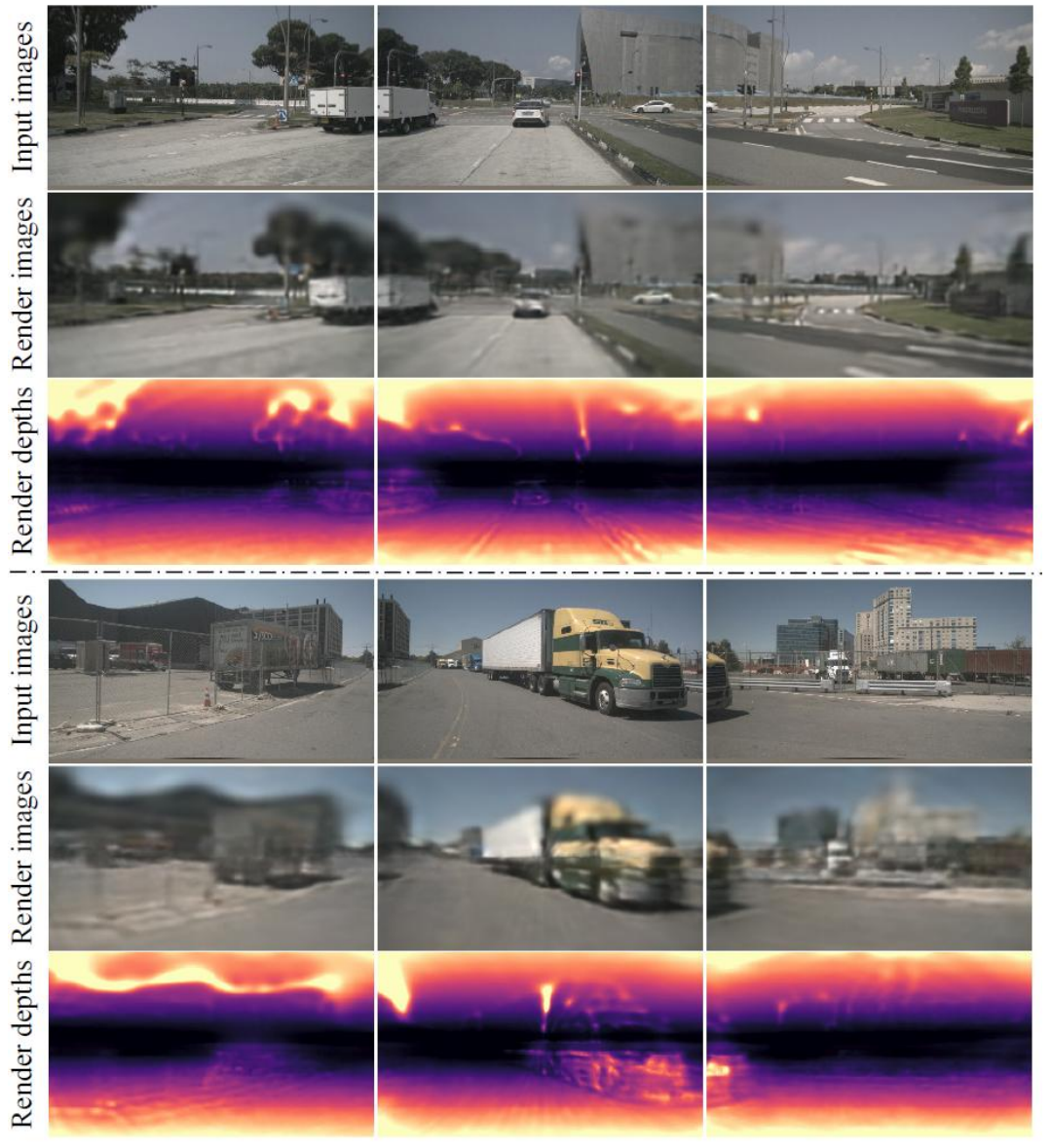


Occupancy Prediction

Occupancy Ground Truth

Visualization

Rendered Results



Conclusion

VisionPAD: A Vision-Centric Pre-training Paradigm for Autonomous Driving:

- We propose the first vision-centric pre-training method which relies solely on images as pre-trained supervision;
- The introduced 3DGS representation and proposed photometric consistency and self-supervised velocity estimation module for multi-frame and multi-view image reconstruction, greatly boosting the performance of camera-based algorithms on three tasks;
- The proposed VisionPAD pre-training paradigm achieves **impressive** performance on the **Occ3D** and **nuScenes** benchmarks for three perception tasks.

香港中文大學(深圳)
The Chinese University of Hong Kong, Shenzhen

VisionPAD: A Vision-Centric Pre-training Paradigm for Autonomous Driving

Thanks for watching!

Haiming Zhang [1,2], *Wending Zhou* [1,2], *Yiyao Zhu* [3], *Xu Yan* [4] †, *Jiantao Gao* [4], *Dongfeng Bai* [4], *Yingjie Cai* [4], *Bingbing Liu* [4], *Shuguang Cui* [2,1], ***Zhen Li*** [2,1]†

[1] The Future Network of Intelligence Institute, The Chinese University of Hong Kong (Shenzhen),
[2] School of Science and Engineering, The Chinese University of Hong Kong (Shenzhen),
[3] HKUST
[4] Huawei Noah's Ark Lab