7 Appendix

7.1 Convergence Analysis for View Attention

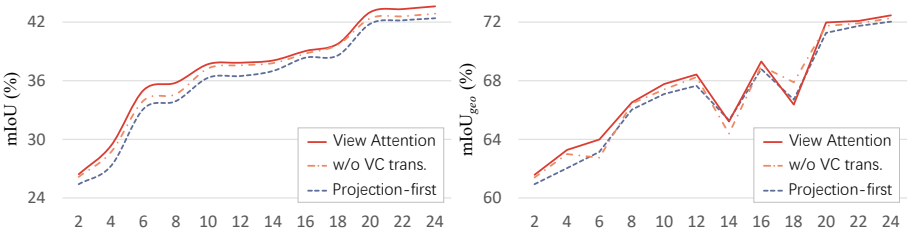

Fig. 8: Convergence Analysis. Our view attention achieves not only faster convergence but also impressive performance. All models are trained for 24 epochs.

Fig. 8 illustrates the convergence analysis of the spatial interaction module, where we replace our view attention with the projection-first interaction method as BEVFormer [18]. The results showcase that our view attention achieves not only faster convergence but also impressive performance. The term "w/o VC trans." denotes learning 3D sample points in the ego-centric perception coordinate system without view angle rotation, highlighting the insightful rotation invariance introduced by our view attention.

7.2 Comparison of Temporal Modeling

Table 8: Comparison of Temporal Modeling. "**SW**" denotes sliding window training approach.

Operation	Train	Time (h)	mIoU	IoU$_{geo}$	mAVE↓
DeformAttn.	**SW**	36	42.17	72.14	0.417
DeformAttn.	Video	13	**42.54**	**72.36**	0.**412**
CNN	Video	13	41.95	71.73	0.424

ViewFormer is trained and tested with online streaming video, while the baseline [18] is trained with a local sliding window and tested with online streaming video. The results in Tab. 8 demonstrate that maintaining consistency between training and inference leads to improved accuracy. It is noteworthy that

the sliding window training approach is time-consuming due to redundant re-inference for each frame window. Benefiting from the streaming memory mechanism [26,35], our ViewFormer pushes each frame into the memory queue for subsequent temporal interaction, thus significantly reducing the training time. We also replace our transformer-based operation, *i.e.* the deformable attention [42], with a CNN-based temporal interaction operation as in [19,26], which yields suboptimal results.

7.3 Depth Supervision Analysis

Table 9: Depth Supervision Analysis for 3D Occupancy Prediction on Occ3D benchmark.

Method	Backbone	Depth	mIoU	IoU_{geo}	others	barrier	bicycle	bus	car	const. veh.	motorcycle	pedestrian	traffic cone	trailer	truck	driv. surf.	other flat	sidewalk	terrain	manmade	vegetation
FB-OCC [19]	InternT	✓	38.69	69.95	11.4	41.42	24.27	46.01	49.38	**24.56**	27.06	28.09	25.61	32.23	38.46	80.97	42.99	50.95	56.15	40.55	37.61
ViewFormer (ours)	InternT	✓	**43.61**	**72.46**	**13.82**	**50.32**	**29.49**	**49.24**	**54.52**	24.34	**32.72**	**31.09**	**31.49**	**34.44**	**41.62**	**85.47**	**51.27**	**59.03**	**62.15**	**48.33**	**42.06**
FB-OCC [19]	InternT	✗	36.26	67.48	10.71	37.06	23.55	42.60	47.16	19.10	25.94	25.75	23.52	29.91	35.60	79.73	41.41	49.51	54.50	35.58	34.23
ViewFormer (ours)	InternT	✗	**41.76**	**71.37**	**12.98**	**47.70**	**27.85**	**45.12**	**53.02**	**21.67**	**28.93**	**28.80**	**29.67**	**31.02**	**39.21**	**86.06**	**51.22**	**58.94**	**62.00**	**45.50**	**40.18**

As mentioned in Sec. 5.2 of our paper, we follow FB-OCC [19] to adopt depth supervision in backbone training for a fair comparison (note that depth supervision is not adopted in Tab. 2). Now, let's delve into the detailed effects of depth supervision. In Tab. 9, we present 3D occupancy results with depth supervision disabled. In the absence of depth supervision, the performance gap between our ViewFormer and FB-OCC widens: our ViewFormer outperforms FB-OCC by 5.5 mIoU and 3.89 mIoU_{geo}. It is evident that without depth supervision, our model shows a modest drop of 1.09 mIoU_{geo} (72.46 vs. 71.37), while FB-OCC presents a larger drop of 2.47 mIoU_{geo} (69.95 vs. 67.48). Due to the limited availability of depth data in vision-centric datasets, our method, with lower reliance on depth information, demonstrates superior generality.

7.4 Two-DOF View Attention

In Sec. 3.1, we present a single degree of freedom (DOF) version view attention, correlated to a query's view angle. As illustrated in Fig. 9, our two-DOF view attention involves an additional altitude angle. Similar to Eq. (2), the query-related altitude angle $\boldsymbol{\phi}$ and the corresponding rotation matrix $\boldsymbol{R}(\phi)$ around the y-axis are calculated as:

$$\begin{aligned} \phi &= \arctan2\left(z, \sqrt{x^2+y^2}\right), \\ \boldsymbol{R}(\phi) &= \begin{bmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{bmatrix}. \end{aligned} \tag{7}$$

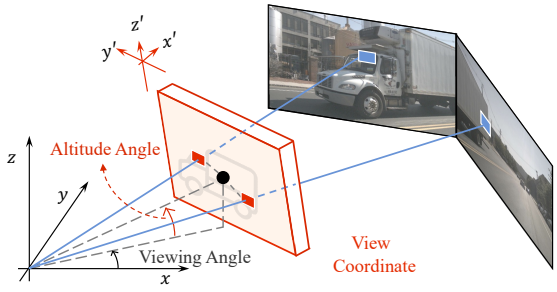

Fig. 9: Two-DOF view attention dealing with both the view angle and altitude angle.

Dealing with both the view angle and altitude angle, Eq. (3) can be extended as:

$$\boldsymbol{p_s} = \boldsymbol{p} + \boldsymbol{R}(\theta) \cdot \boldsymbol{R}(\phi) \cdot \Delta \boldsymbol{p}^{\mathcal{T}}. \tag{8}$$

In contrast to our single-DOF version dealing with only the view angle, the two-DOF version actually brings no performance gain. The reason is considered to lie in the fact that like the nuScenes [5, 9] dataset, the cameras are almost horizontally mounted in an AD system, our single-DOF version is therefore capable of representing the rotation invariance. In other scenarios such as the field of 3D reconstruction where the cameras are irregularly distributed, however, the two-DOF version presents a valuable option.

7.5 Discussion about the Necessity of Occupancy Flow

In our paper, we create our occupancy flow benchmark FlowOcc3D, to explore the potential of fine-grained representation of dynamic scenes. Questions may arise regarding how the integrity of objects is maintained in such representation. We would like to emphasize that the occupancy-level flow representation tackles challenges faced in the traditional object-level flow representation, *e.g.*, lack of effective representation for irregularly shaped objects and background information. Existing occupancy tasks do not address the object integrity issue, but rather resembles semantic segmentation in 3D space. The object integrity is more similar to the task handled in instance segmentation, which is in fact out the scope of occupancy tasks.