Supplementary Material

Jie Cheng[1] Xiaodong Mei[1] Ming Liu[1,2]
HKUST[1] HKUST(GZ)[2]
{jchengai, xmeiab}@connect.ust.hk, eelium@ust.hk

A. Implementation Details

Training. For all experiments, we train the model using an AdamW [5] Optimizer with a weight decay of 1e-4 and batch size of 128 on 4 GPUs. We use cosine learning rate decay and the intial learning rate is 1e-3. The dropout rate in all transformer blocks is set to 0.2. We use an agent-centric coordinates system and only consider agents and lane segments within 150 meters of the focal agent. The latent feature dimension is set to 128.

Agent embedding. The agent's embedding layer is a Feature Pyramid Network (FPN), primarily composed of neighborhood attention blocks (NATBlock) and 1D-convolution networks, depicted in Figure 1. The agent's input is of the shape $N \times 50 \times 4$, which corresponds to a sequence of historical states spanning 5 seconds, sampled at a frequency of 10 Hz. Each state includes the agents' displacement and velocity difference relative to the previous timestamp, along with a padding flag indicating the observation status. The NATBlock exhibits an identical structure to the standard Transformer encoder block [7] (multi-head self-attention, add & norm, and fully-connected layer), except for the replacement of self-attention with 1D neighborhood attention [4]. All downsample and upsample operators are implemented with 1D-convolution, having a down/up-sampling ratio of 2. We employ the same layer (but a seperate one) for the agents' future embedding during the pre-training phase.

Lane embedding. The non-overlapping lane segments are acquired using the official Argoverse 2 API[1]. Each individual lane segment is precisely interpolated to consist of 20 points. Each point encompasses its two-dimensional coordinates, normalized with respect to its geometrical center, as well as a padding flag denoting its presence within the region of interest to the focal agent. The architecture of the lane embedding layer adheres to the PointNet design [6], with a comprehensive depiction of its detailed structure provided in Figure 2.

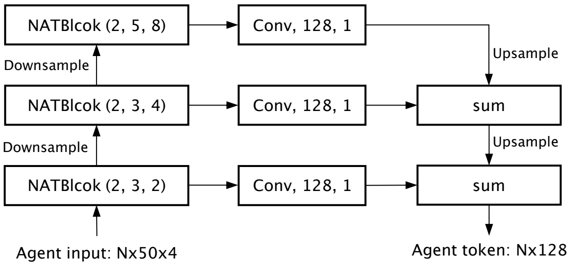


Figure 1: Detailed architecture of the agent history embedding layer. Numbers in the *NATBlock* denote the number of stacked blocks, the kernel size of neighborhood attention, and the number of heads. Numbers in the *Conv* indicate the hidden dimensions and stride.

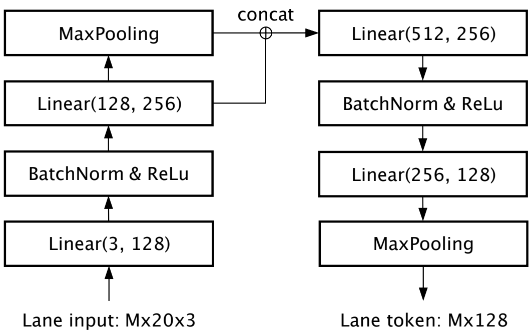


Figure 2: Detailed architecture of the lane embedding layer.

Fine-tune. We employ an end-to-end finetuning approach for the motion forecasting task, as illustrated by the overall architecture depicted in Figure 3. Throughout the fine-tuning process, only the history and lane features are embedded as inputs. Subsequently, the encoded history tokens of the agents are utilized for generating future predictions and associated confidences via the multi-modal decoder.

Experiment Setting. We report the default setting for the pre-training and fine-tuning phase of Forecast-MAE in Table 1 and Table 2.

SSL-Lanes. To adapt SSL-Lanes [1] to the Argoverse 2 dataset, we change the history and future length to 50 and 60, respectively. The region of interest is increased from 100 to 150 meters accounting for the longer observa-

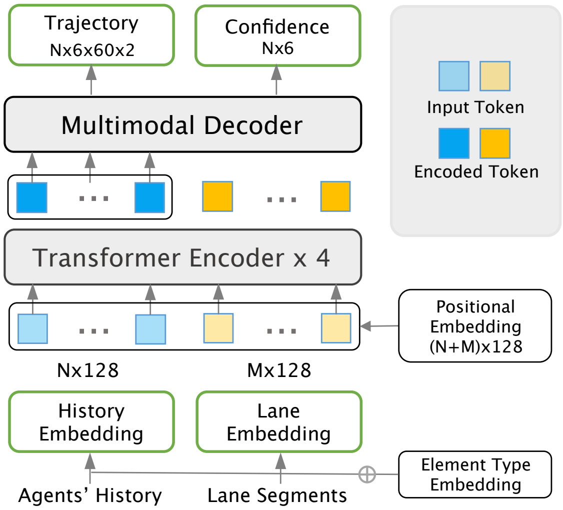


Figure 3: Overall architecture of fine-tune model

tion/prediction horizon. We follow the default experiment setting of SSL-Lanes, shown in Table 3.

config	value
optimizer	AdamW
learning rate	1e-3
weight decay	1e-4
learning rate schedule	cosine
batch size	128
training epochs	60
warmup epochs	10
masking ratio	[0.4, 0.5]
loss weight	[1.0, 1.0, 0.35]
augmentation	none

Table 1: Experiment setting for Forecast-MAE pre-training. Masking ratios refer to history trajectory and lane segments, respectively.

B. Additional Results

Results on more datasets. We provide preliminary experimental results on Argoverse 1 [2] and WOMD [3]. The results are shown in Table 4.

More visual results and comparisons. We compare the performance of Forecast-MAE with two baselines, namely *SSL-Lanes* and *Scratch* (trained from scratch). The comparative visualization results are displayed in Figure 4. In comparison to the baselines, our Forecast-MAE model yields greater accuracy in direction and velocity prediction, even in high-speed and highly interactive scenarios. Notably, our fine-tuned model is the only one that captures lane-change behavior in scene (III). Furthermore, Forecast-MAE can generate a diverse range of multi-modal predictions while simultaneously ensuring precision, whereas other methods often predict infeasible trajectories. The visualization outcomes provide compelling evidence that our method is highly effective in encapsulating motion, road geometry, and cross-modal interaction features.

config	value
optimizer	AdamW
learning rate	1e-3
weight decay	1e-4
learning rate schedule	cosine
batch size	128
training epochs	60
warmup epochs	10
augmentation	none

Table 2: Experiment setting for Forecast-MAE fine-tuning

config	value
optimizer	Adam
learning rate	1e-3
learning rate schedule	1e-4 at 62
batch size	128
training epochs	80
augmentation	none

Table 3: Experiment setting for SSL-Lanes.

Dataset	Method (year)	minADE	minFDE	MR
WOMD (test set)	DenseTNT('21)	1.039	1.551	0.157
	MTR('23)	0.605	1.225	0.137
	Ours/scratch	0.689	1.341	0.182
	Ours/fine-tune	0.632	1.253	0.167
AV1 (val set)	TPCN('21)	0.73	1.15	0.11
	Autobots('22)	0.73	1.10	0.12
	Ours/scratch	0.736	1.103	0.103
	Ours/fine-tune	0.710	1.054	0.945

Table 4: Preliminary results on WOMD test set and Argoverse 1 validation set, and comparison with representative methods. All results are from single model (w/o ensemble).

Maksed scene reconstruction. We showcase the reconstruction results of two complex scenarios from Argoverse 2 validation set using our pre-trained model, which was trained with a history and lane masking ratio of 0.5. As depicted in Figure 5 (first row), the pre-trained model exhibits a remarkable ability to recover the original scenario,

[1] https://github.com/argoverse/av2-api

including the history and future trajectories of agents and intricate lane geometries. Interestingly, our model performs well even with higher lane masking ratios (second and third rows in Figure 5). Despite a high lane masking ratio of 0.8, where most of the lane structures are lost, our model can still reconstruct most of the lane structures reasonably. These results suggest that our model has learned rich and profound scene representations through MAE-based self-supervised pre-training.

References

[1] Prarthana Bhattacharyya, Chengjie Huang, and Krzysztof Czarnecki. Ssl-lanes: Self-supervised learning for motion forecasting in autonomous driving. In *6th Annual Conference on Robot Learning*. 1

[2] Ming-Fang Chang, John Lambert, Patsorn Sangkloy, Jagjeet Singh, Slawomir Bak, Andrew Hartnett, De Wang, Peter Carr, Simon Lucey, Deva Ramanan, et al. Argoverse: 3d tracking and forecasting with rich maps. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8748–8757, 2019. 2

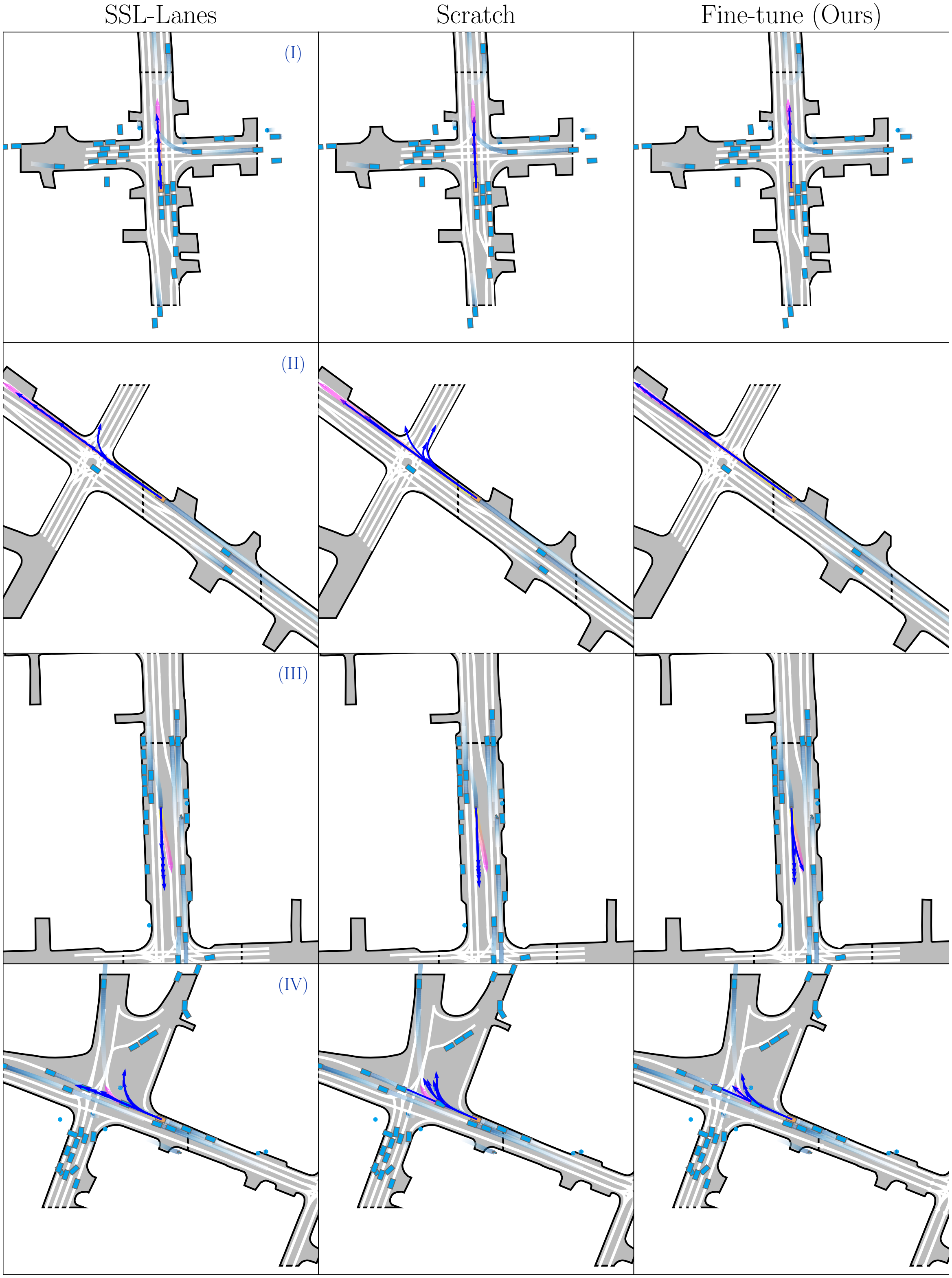
[3] Scott Ettinger, Shuyang Cheng, Benjamin Caine, Chenxi Liu, Hang Zhao, Sabeek Pradhan, Yuning Chai, Ben Sapp, Charles R Qi, Yin Zhou, et al. Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9710–9719, 2021. 2

[4] Ali Hassani, Steven Walton, Jiachen Li, Shen Li, and Humphrey Shi. Neighborhood attention transformer. *arXiv preprint arXiv:2204.07143*, 2022. 1

[5] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 1

[6] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 1

[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 1


SSL-Lanes
Scratch
Fine-tune (Ours)
(I)
(II)
(III)
(IV)

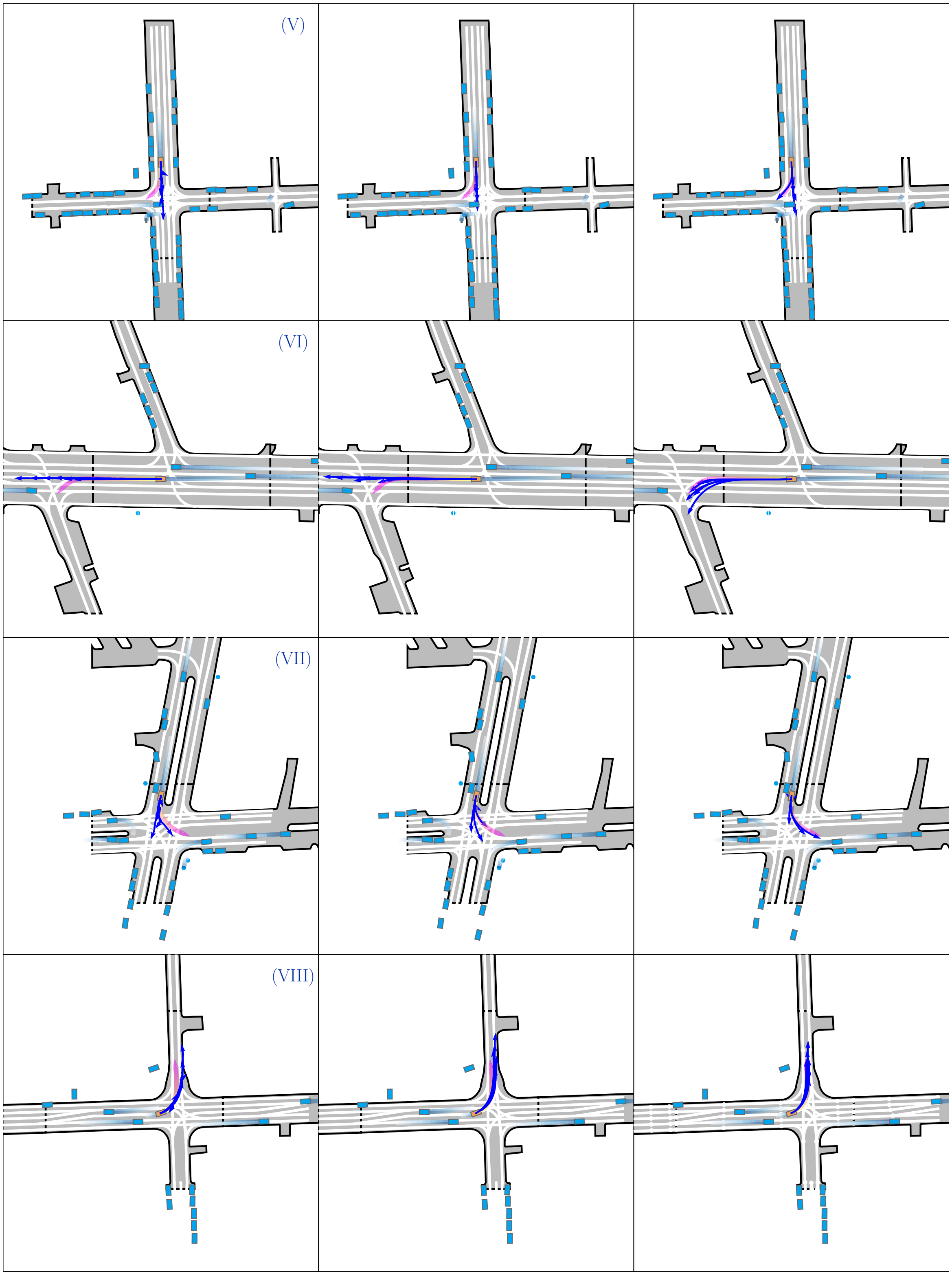
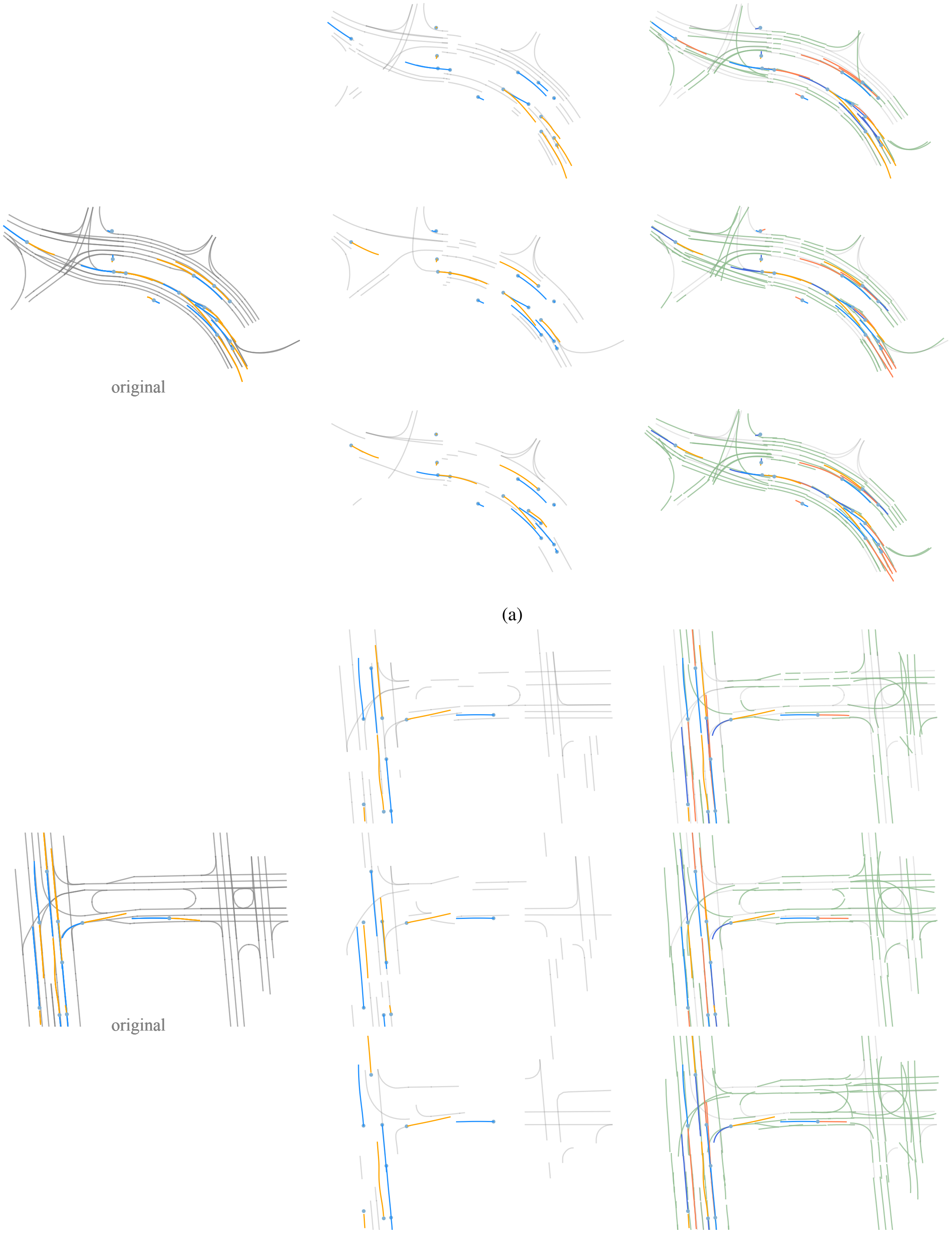


Figure 4: **More visualization results and comparisons on Argoverse 2 *validation* set.** The focal agent is denoted in orange, while the others are indicated in blue. The deep blue lines with arrows denote the predictions, and the gradual pink lines with arrows represent the ground truth. The arrows indicate the direction of motion. The gradual blue lines represent the historical trajectories, with the color transitioning from light to dark to indicate the direction of motion.



(a)

(b)

Figure 5: **Masked scene reconstructions results on Argoverse 2 *validataion* set.** The pre-trained model with a history/lane masking ratio of 0.5 is utilized to process input scenarios with higher lane masking ratios. The inputs and results are displayed in a top-down sequence, corresponding to the lane masking ratios of 0.5, 0.65, and 0.8.