StreamingFlow: Streaming Occupancy Forecasting with Asynchronous Multi-modal Data Streams via Neural Ordinary Differential Equation

Supplementary Material

1. More model designs

1.1. Preliminary

Neural-ODE fundamentals. The Neural Ordinary Differential Equation (N-ODE) is proposed for learning some simple continuous implicit processes in dynamic systems. The core characteristic is that it uses neural networks to parameterize derivatives of hidden states $f(h(t), t, \theta)$ instead of specifying a discrete sequence of hidden layers $h(t)$. Given a continuous process h_t, the update process is to calculate the derivatives $\frac{dh(t)}{dt}$ via a neural network,

$$h(t+1) = h_t + f(h_t, \theta_t) \tag{1}$$

$$\frac{dh(t)}{dt} = f(h(t), t, \theta) \tag{2}$$

1.2. Decoders, Loss and Post-processing

The decoders are inherited from FIERY[1, 3, 4]. Five distinct decoders produce centerness regression, BEV segmentation, offset to the centers, future flow vectors, and instance given the BEV feature representation of past and future frames. The shared BEV feature will be input into a shared Resnet18 BEV backbone and five independent CNN blocks.

The loss design consists of spatial regression loss, segmentation loss, and probabilistic loss. Spatial regression loss is responsible for regressing centerness and offsets in a L1 loss or mean square error(MSE/L2) loss manner. Segmentation loss is the computation of the cross-entropy loss on multi-frame BEV semantic grids from the past to the future. Probabilistic loss computes the divergence between updated BEV features and measurement features with regard to their mean and variance on BEV grids. The overall loss is calculated as follows:

$$Loss = \lambda_1 * L_{seg} + \lambda_2 * L_{spatial} + \lambda_3 * L_{kld} \tag{3}$$

In Eq. (3), λ is the weight of each loss. For simplicity, we set $\lambda_1 = 0.5$, $\lambda_2 = 0.5$, and $\lambda_3 = 1.0$.

2. More Experiments

2.1. Runtime Analysis

Table. 1 compares the run-time training memory of the proposed method with baselines in different dataset settings. As the time interval becomes denser, the training cost for standard GRU units becomes unaffordable, whereas the proposed method is more adaptable. For standard 4-keyframe supervision, StreamingFlow only requires $+2G$ more memory compared to a BEVFusion-style implementation. As the density of supervision signals increases, the ODE approach requires $-4G$ less memory than standard GRU counterparts.

The inference speed of StreamingFlow is measured by the average time required to process validation samples over 250 forward passes on a laptop equipped with a single RTX3090. As StreamingFlow inherits the same framework and modules from FIERY[3], it is compared with FIERY[3] and StretchBEV[1]. For the settings of the standard task and variable ode steps, StreamingFlow runs at 0.1968s/sample, faster than FIERY(0.6436s/sample) and StretchBEV(0.6469s/sample) reported in [1]. The SpatialGRU-ODE works at a similar speed with prior temporal modules. The inference speed for tasks with finer granularity (40-frame experiment) is around 0.5s per sample. Obviously, higher prediction frequencies harm the runtime delay. Therefore, sparse streaming prediction based on variable ODE step by request is recommended.

Dataset	Config	Supervised frames	Memory
nuScenes	GRU-base	4	11G
nuScenes	GRU-ODE	4	13G
nuScenes	GRU-base	40	39G
nuScenes	GRU-ODE	40	OOM
Lyft	GRU-base	10	28G
Lyft	GRU-ODE	10	24G

Table 1. Runtime analysis of training cost of different model configs. 'OOM' denotes out-of-memory for one batch in a single A6000 GPU (48G memory)

2.2. Baselines

Baselines from prior works Vision track. FIERY[3] is the first practice for end-to-end stochastic occupancy flow prediction. StretchBEV[1][1] uses a variational autoencoder for learning implicit temporal dynamics and future prediction in a decoupled style. ST-P3[4] is the first end-to-end planning framework that considers occupancy prediction. BEVerse[14] is the first multi-task model for both object- and grid-level perception.

[1] StretchBEV-P uses ground-truth labels of past frames as a posterior for prediction, which is unfair for comparison with end-to-end occupancy prediction, so only StretchBEV without labels is compared in the table.

LiDAR track. MotionNet[12] is the first practice for learning BEV grid motion using a simple spatial-temporal voxel-based backbone (STPN). BE-STI[11] develops the backbone with two new blocks, SeTE and TeSe to enhance temporal feature representation. Since LiDAR track algorithms are not originally proposed for this task, we reimplement them using original BEV backbones and prediction heads for this task.

Fusion track. FusionAD[13] is a multi-modality, multi-task, end-to-end driving framework. They build a transformer to conduct multi-modal BEV-level fusion and downstream perception, prediction and planning tasks. Occupancy prediction results are from the original paper.

Baselines of fusion strategies Synchronous fusion. The process is first multi-modal spatial fusion, then mixed-modal temporal fusion, and finally standard GRU modules. The assumption is that, at each frame, LiDAR points are tightly synchronized and fused with the nearest images. Spatial fusion follows the same methodology as BEVFusion[5]. This process requires strict synchronization and weak time interval uniformity.

Asynchronous fusion: The process is first single-modal temporal fusion, then standard GRU modules, and finally mixed-modal spatial fusion on future timestamps. Single-modal temporal fusion is first performed using the spatio-temporal convolution (STC) unit, and then spatial fusion is performed during prediction. The assumption is that the perception and prediction times are strictly uniform. This process requires only weak synchronization and strict time interval uniformity.

2.3. Perception Results

As a similar task to flow prediction, we also compare the performance of BEV segmentation of intermediate representations with prior arts in Tab. 2. We evaluate two main traffic agent categories, vehicle, and pedestrian by IoU metric. With timestamp-agnostic camera-LiDAR fusion by SpatialGRU-ODE, StreamingFlow also achieves impressive progress in typical agent segmentation. It surpasses ST-P3[4] by +10.7 for vehicles and +22.7 for pedestrians.

2.4. Analysis and Discussion

We provide an intuitive analysis of streaming forecasting training and inference efficiency. Either dense or sparse labels are applicable for the supervised signal of future frames, but the extremely sparse supervised signal as supervision may degrade the performance of SpatialGRU-ODE, as the state may be updated too many times until the next supervised frame. In contrast, the denser supervised signal can strengthen the method to surpass the state-of-the-art synchronized spatial fusion method. For accurate inference at future timestamps with low latency, SpatialGRU-ODE with variable time step, which is closely related to different prediction requests, is the best trade-off for accuracy and latency.

Method	Vehicle / IoU	Pedestrian / IoU
VED[6]	23.3	11.9
VPN[7]	28.2	10.3
PON[9]	27.9	13.9
Lift-Splat[8]	31.2	15.0
IVMP[10]	34.0	17.4
FIERY[3]	38.0	17.2
ST-P3 [4]	40.1	14.5
StreamingFlow	**50.8**	**37.2**

Table 2. Comparison with the state-of-the-art methods for BEV segmentation of vehicles and pedestrians on nuScenes[2] validation set.

2.5. More visualizations

Demo videos for standard occupancy forecasting, streaming occupancy forecasting and long-term zero-shot or supervised occupancy forecasting for more scenarios will be available at `https://github.com/synsin0/StreamingFlow`.

In the 40-frame prediction visualization, the static instances remain static overall with only minor changes in the perception range. Though grid-centric perception is discrete in essence, the prediction results show that StreamingFlow successfully learns the temporal dynamics in continuous time series.

References

[1] Adil Kaan Akan and Fatma Güney. Stretchbev: Stretching future instance prediction spatially and temporally. In *European Conference on Computer Vision*, pages 444–460. Springer, 2022. 1

[2] Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11618–11628, 2020. 2

[3] Anthony Hu, Zak Murez, Nikhil Mohan, Sofía Dudas, Jeffrey Hawke, Vijay Badrinarayanan, Roberto Cipolla, and Alex Kendall. FIERY: Future Instance Prediction in Bird's-Eye View from Surround Monocular Cameras. *Proceedings of the IEEE International Conference on Computer Vision*, pages 15253–15262, 2021. 1, 2

[4] Shengchao Hu, Li Chen, Penghao Wu, Hongyang Li, Junchi Yan, and Dacheng Tao. St-p3: End-to-end vision-based autonomous driving via spatial-temporal feature learning. In *European Conference on Computer Vision*, pages 533–549. Springer, 2022. 1, 2

[5] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang,

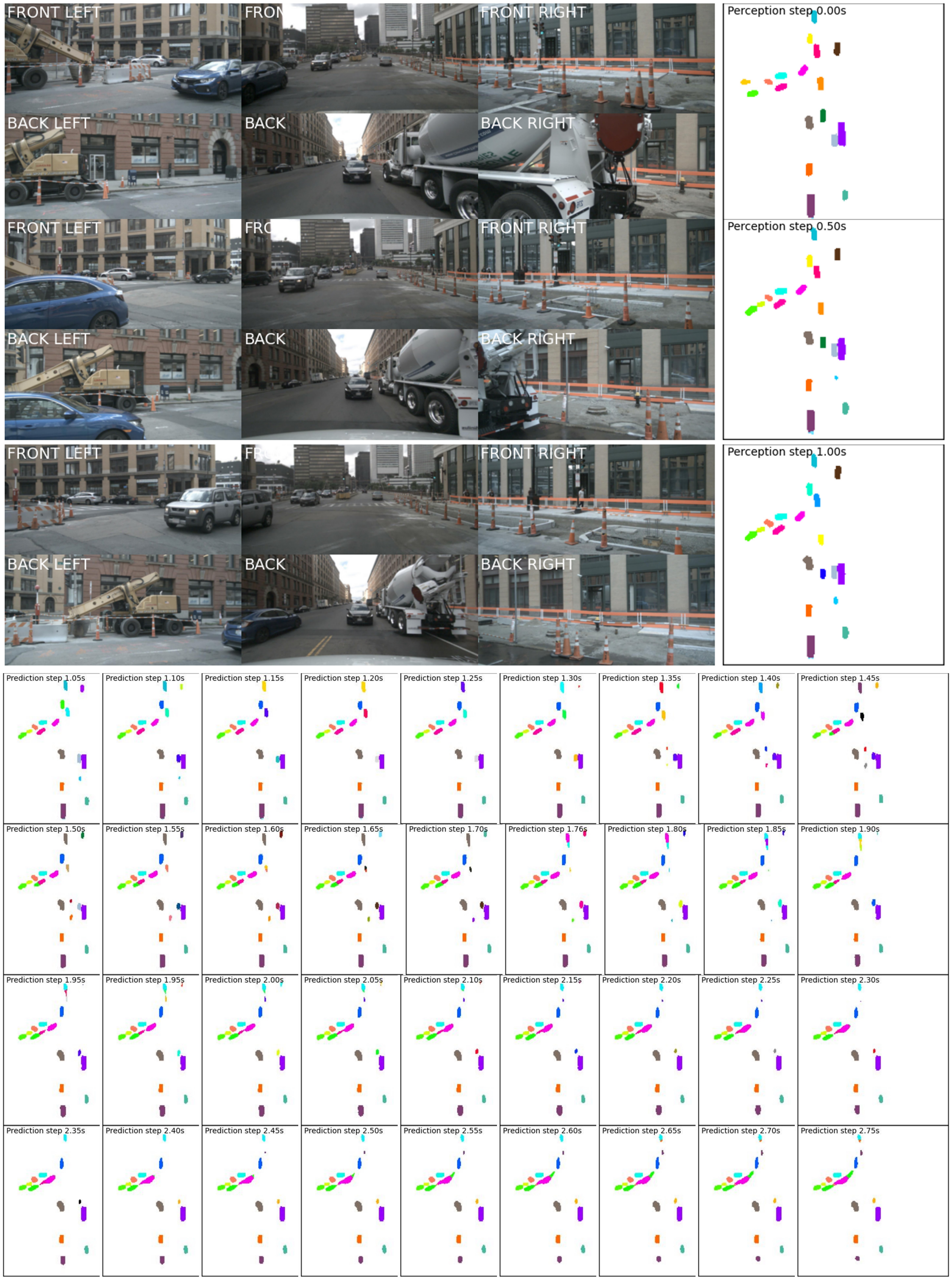

Figure 1. Visualization of StreamingFlow for continuous panoptic occupancy flow prediction at a busy intersection. Given 3 key-frame cameras inputs and asynchronous 5 key-frame LiDAR inputs, we are able to accurately predict the continuous trend of dynamic trend(front left view) only with sparse supervision.

Huizi Mao, Daniela L Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *2023 IEEE international conference on robotics and automation (ICRA)*, pages 2774–2781. IEEE, 2023. 2

[6] Chenyang Lu, Marinus Jacobus Gerardus van de Molengraft, and Gijs Dubbelman. Monocular Semantic Occupancy Grid Mapping With Convolutional Variational Encoder–Decoder Networks. *IEEE Robotics and Automation Letters*, 4(2):445–452, 2019. 2

[7] Bowen Pan, Jiankai Sun, Ho Yin Tiga Leung, Alex Andonian, and Bolei Zhou. Cross-view semantic segmentation for sensing surroundings. *IEEE Robotics and Automation Letters*, 5(3):4867–4873, 2020. 2

[8] Jonah Philion and Sanja Fidler. Lift, Splat, Shoot: Encoding Images from Arbitrary Camera Rigs by Implicitly Unprojecting to 3D. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 12359 LNCS:194–210, 2020. 2

[9] Thomas Roddick and Roberto Cipolla. Predicting semantic map representations from images using pyramid occupancy networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11135–11144, 2020. 2

[10] Hengli Wang, Peide Cai, Yuxiang Sun, Lujia Wang, and Ming Liu. Learning interpretable end-to-end vision-based motion planning for autonomous driving with optical flow distillation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 13731–13737. IEEE, 2021. 2

[11] Yunlong Wang, Hongyu Pan, Jun Zhu, Yu-Huan Wu, Xin Zhan, Kun Jiang, and Diange Yang. Be-sti: Spatial-temporal integrated network for class-agnostic motion prediction with bidirectional enhancement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17093–17102, 2022. 2

[12] Pengxiang Wu, Siheng Chen, and DImitris N. Metaxas. MotionNet: Joint Perception and Motion Prediction for Autonomous Driving Based on Bird's Eye View Maps. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 11382–11392, 2020. 2

[13] Tengju Ye, Wei Jing, Chunyong Hu, Shikun Huang, Lingping Gao, Fangzhen Li, Jingke Wang, Ke Guo, Wencong Xiao, Weibo Mao, et al. Fusionad: Multi-modality fusion for prediction and planning tasks of autonomous driving. *arXiv preprint arXiv:2308.01006*, 2023. 2

[14] Yunpeng Zhang, Zheng Zhu, Wenzhao Zheng, Junjie Huang, Guan Huang, Jie Zhou, and Jiwen Lu. Beverse: Unified perception and prediction in birds-eye-view for vision-centric autonomous driving. *arXiv preprint arXiv:2205.09743*, 2022. 1