Motion Prediction for Autonomous Driving with Drone Traffic Monitoring

Weijiang Xiong[1*], Alexandre Alahi[2], and Nikolas Geroliminis[1]

[1]Urban Transport Systems Laboratory (LUTS), École Polytechnique Fédérale de Lausanne (EPFL), Switzerland

[2]Visual Intelligence for Transportation Laboratory (VITA), École Polytechnique Fédérale de Lausanne (EPFL), Switzerland.

Short summary

Motion prediction is essential for safe and reliable decision-making in autonomous driving and robotic systems. Although severe performance degradation is usually expected when a model is deployed to a new environment with different road structures and traffic patterns, collecting onboard driving logs and annotating High-Definition (HD) maps for every new city takes tremendous effort and time. Therefore, this work investigates the potential of using drone-based traffic monitoring as an efficient and scalable data source for transferring motion forecasting models to unseen environments. With occlusion-free drone observations, any traffic participant can be selected as the ego vehicle for training motion forecasting models, yielding high-quality data at much larger volume and with a more diverse distribution. We introduce Songdo Drive, a motion forecasting dataset built from 137 hours of drone traffic monitoring over 20 complex intersections in Songdo, South Korea. Experiments show that a zero-shot model trained on an existing autonomous-driving benchmark degrades significantly when evaluated on Songdo Drive, yet the performance can be substantially improved by leveraging the advantage of drone-based trajectory data.

Keywords: Autonomous Vehicles, Trajectory Prediction, Urban Traffic Monitoring, Drones

1 Introduction

Predicting the future motion of vehicles is a core capability for autonomous driving, because it supports downstream tasks such as risk assessment, interaction modeling, and motion planning. Recent progress has been supported by large-scale benchmarks such as the Waymo Open Motion Dataset, nuScenes, and Argoverse 2, which have enabled strong deep learning models to emerge (Ettinger et al., 2021; Caesar et al., 2020; **?**). These datasets are expensive assets collected from vehicles with extensive sensors over a long driving operation time. Therefore, such high-standard data properties are quite challenging to replicate in a new city.

At the same time, the out-of-distribution problem of deep learning models has been a long-standing pain point (Zhou et al., 2023). A model trained with data from one city may face variations in traffic organization, infrastructure layout and behavioral patterns when deployed elsewhere, which can substantially degrade forecasting performance. For many cities, collecting new onboard driving logs and HD maps is impractical, especially when the goal is to obtain a sufficient amount of data efficiently. This motivates the research on alternative data sources that are easier to acquire while still capturing realistic multi-agent interactions.

In recent years, there has been a growing interest in using drones for traffic monitoring in the traffic flow research community (Barmpounakis & Geroliminis, 2020; Fonod et al., 2025). Compared with onboard sensing, bird's-eye-view videos from drones can cover a larger field of view, suffer less from perspective occlusions, and capture more natural traffic interactions because traffic participants are not distracted by the monitoring setup. Existing drone datasets, e.g., highD, inD, and SinD, have demonstrated the advantage of capturing diverse motion patterns for autonomous driving systems using data from drones (**???**). But these drone datasets remain much smaller in volume and scope than mainstream vehicle-centric benchmarks and are not often used in autonomous-driving research. In contrast, two large-scale datasets collected for traffic monitoring research, i.e.,

pNEUMA and Songdo Traffic, cover much larger urban road networks with challenging interactions, e.g., traffic signals, bus stops, and heavy congestion (Barmpounakis & Geroliminis, 2020; Fonod et al., 2025). They can provide the level of data diversity necessary for training robust models in autonomous driving, but their capabilities have not been explored in the context of motion forecasting.

With this research gap in mind, this work proposes to exploit the potential of drone-based traffic monitoring for transferring motion forecasting models to a new city. Our approach efficiently creates training data for vehicle-centric motion forecasting models from the scene-centric drone monitoring footage, and the resulting dataset covers a wide range of motions in challenging traffic scenarios. In summary, this study is primarily motivated by the following research questions:

1. How to efficiently make drone-captured traffic monitoring data suitable for training and evaluating modern vehicle trajectory prediction models?
2. How well do existing vehicle trajectory prediction methods perform when applied to drone-captured trajectory data beyond the benchmarks they were designed for?

2 Methodology

Although deep learning-based methods have demonstrated state-of-the-art performance on many trajectory prediction benchmarks, they struggle to transfer the learned knowledge from one city to another due to the domain shift problem (Zhou et al., 2023). In this work, we investigate the problem of training motion forecasting models in a city with no dedicated vehicle driving logs or High-Definition Maps. Generalizing into an unseen environment is inherently difficult, and expecting a model to perform well without any adaptation is thus unrealistic. However, instead of collecting onboard sensing data with instrumented vehicles, we show that footage from a parallel research thread, i.e., drone-based traffic monitoring, can be efficiently leveraged to address this challenge.

Drones provide highly accurate, occlusion-free bird's-eye-view (BEV) observations of traffic scenes, capturing all traffic participants and their interactions in a naturalistic way. Therefore, any car present in the drone footage can be selected as the ego vehicle, with the others treated as surrounding vehicles. This yields many more valid data samples from the same monitoring logs for training and evaluation of motion forecasting models. In contrast, onboard autonomous-driving datasets are often limited to the perception range of the ego vehicle, and thus have lower data efficiency.

Problem Description

In the motion forecasting task, a model usually has the following information:

- Map information, e.g., lanes, stop lines, drivable area and traffic light states
- Past trajectories, for the ego vehicle, surrounding vehicles, cyclists and pedestrians.

Then, a model is required to predict multiple possible future trajectories of the ego vehicle, along with corresponding probabilities. Formally, the task can be defined as:

$$\left\{\hat{\mathbf{y}}_{1:T}^{(k)},\, p^{(k)}\right\}_{k=1}^{K} = f_\theta\left(\mathcal{M}, \mathbf{x}_{-H:0}^{\text{ego}}, \mathbf{x}_{-H:0}^{1:N}\right), \quad \text{s.t.} \sum_{k=1}^{K} p^{(k)} = 1 \tag{1}$$

where $\mathbf{x}_{-H:0}^{\text{ego}}$ is the ego-vehicle trajectory over H past steps, $\mathbf{x}_{-H:0}^{1:N}$ are the histories of the N surrounding agents, and $\mathcal{M}$ denotes the map information. The model f_θ predicts K candidate future trajectories for the next T steps together with the probabilities $p^{(k)}$ for all candidates.

Following the practice of UniTraj (Feng et al., 2024), we evaluate the open-loop motion forecasting performance using the following metrics:

- Minimum Average Displacement Error (minADE), the minimum average distance (in meters) between the predicted trajectory and the ground truth over all the T future steps. The minimum is taken over the K predicted trajectories.

- Minimum Final Displacement Error (minFDE), the minimum distance (in meters) between the predicted position and the true position at the T-th time step. Similar to minADE, the minimum is also taken over the candidate trajectories.
- Miss Rate, the percentage of samples where minFDE exceeds a certain threshold (e.g., 2 m).
- BrierFDE, the minFDE with a penalty term $(1-p)^2$, where p is the probability of the best predicted trajectory. This metric encourages the model to assign high confidence to accurate predictions.

Data Processing Method

In practice, motion forecasting data is often organized as short-term traffic scenes, which comprise vehicle trajectories (e.g., ego and background vehicles) and map information (e.g., lane layout and traffic lights) (Li et al., 2023). On the other hand, drone-based traffic monitoring typically offers vehicle trajectories over much longer time horizons (Barmpounakis & Geroliminis, 2020). Therefore, proper adaptation is necessary to connect these two lines of work.

Concretely, in this work, we adapt the Songdo Traffic Dataset (Fonod et al., 2025) to the motion forecasting framework UniTraj (Feng et al., 2024), and we coin the resulting dataset as Songdo Drive. The Songdo Traffic dataset provides high-quality geo-referenced locations and speeds at 30 Hz for multiple types of vehicles, accurate size estimates, as well as precise bounding box annotations for the lanes. Meanwhile, UniTraj expects a scene at 10 Hz with 2 seconds of history trajectories, a 1-frame current position, and 6 seconds of future trajectories, which makes 81 frames in total. UniTraj represents the map as polylines, e.g., lane centerlines and stop lines. Figure 1 shows the workflow for scene adaptation.

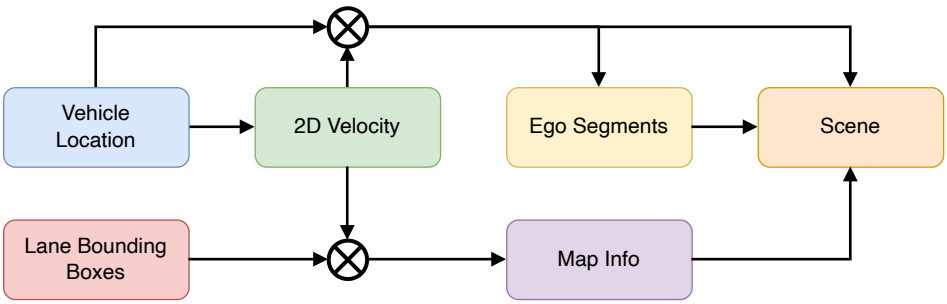


Figure 1: Overview of the Songdo preprocessing pipeline. Trajectories are first used to estimate 2D velocity, including heading and speed, and then to extract fixed-length ego segments. The lane bounding boxes and velocity data are used to derive map information; both branches are then merged into the final scene representation.

Firstly, we estimate 2D velocity for each vehicle. The geographic locations in local coordinates (EPSG:5186) are smoothed by taking a 5-frame centered moving average. Then, the speeds over the X and Y axes are obtained as the temporal gradients of the smoothed locations. Since the heading of a low-speed vehicle is often subject to noise, we filter out the heading estimation if the speed falls below 3 m/s, and fill those invalid headings with valid neighboring headings. The estimated 2D velocity is combined with the initial vehicle locations and sizes to form the trajectory information in the traffic scene.

The combined trajectory information is also used to determine the ego segments, i.e., the car chosen as the ego vehicle and the corresponding 8-second time window used to include vehicle trajectories. We first take candidate 8-second segments with 4-second stride for each car trajectory, such that the 2-second input window can be well-separated to create more diverse samples. The trajectories that are too short to fit one segment are discarded in this step. Then, we apply aggressive filtering to ensure the quality of the retained segments, which rejects the segment if the ego trajectory has missing locations, invalid sizes, invalid velocities or high similarities with a selected segment. The similarity filter compares time overlap and translation-normalized trajectory shape to remove near-duplicate segments, such as vehicles moving closely in leader-follower or parallel patterns.

We derive map information using the lane bounding boxes and the velocity data. For the lane direction, we first take the longer midline of the lane box, and choose the direction that is more

aligned with the observed vehicle speed within the lane bounding box. If the motion evidence is unclear, we leave the lane to be undirected. Then, we place stop line candidates downstream of the directed lane centerlines, and filter out the candidates that are too close to the upstream of another lane. As for the drivable area, we discretize the map into a 1-meter grid, and mark the grid cells that are occupied by any vehicle trajectory as drivable.

Finally, the scene is composed of the trajectory information, chosen ego vehicle, and corresponding map structure.

3 Results and Discussions

In this section, we first present the statistics of Songdo Drive and then benchmark the performance of representative motion forecasting methods.

Dataset Statistics

The monitoring sessions in the Songdo Traffic Dataset provide four days of traffic data over 20 intersections, each day containing 10 monitoring sessions of 30 minutes. Therefore, there are 800 monitoring sessions in total, capturing 137.2 hours of vehicle trajectories. After our processing, we obtain 389K valid vehicles and 750K ego segments, which add up to 1665.8 hours of total segment time. As a result, we achieve a 12x increase compared with the original monitoring time, even after aggressively filtering out 281K trajectory segments to ensure data quality. In our experiments, we split the dataset into 680 training sessions and 120 test sessions, while ensuring that all intersections are in both splits. These statistics are summarized in Table 1 for better clarity.

Table 1: Dataset statistics for the train and test splits, as well as the whole dataset. Original time and segment time are reported in hours, converted from the provided minute counts. Filtered segments are the sum of all filtering categories.

Split	Sessions	Ego segs.	Valid veh.	Time (h)	Seg. time (h)	Filtered
Train	680	635107	331383	116.23	1411.35	241559
Test	120	114507	57860	20.97	254.46	40116
Overall	800	749614	389243	137.20	1665.81	281675

Figure 2 shows the composition of the filtered segments across all exclusion categories, where most segments are filtered out because of duplicate patterns, followed by those too short to fit one segment. In fact, since the vehicle locations are first obtained by object detection and tracking, some trajectories contain missing detections or inaccurate size estimates due to partial observability, and that is the main reason for the remaining filtered segments.

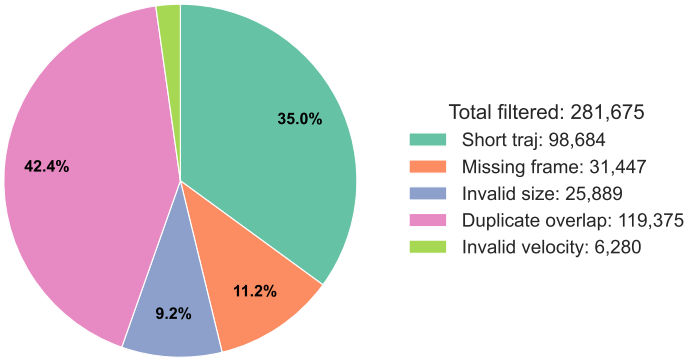
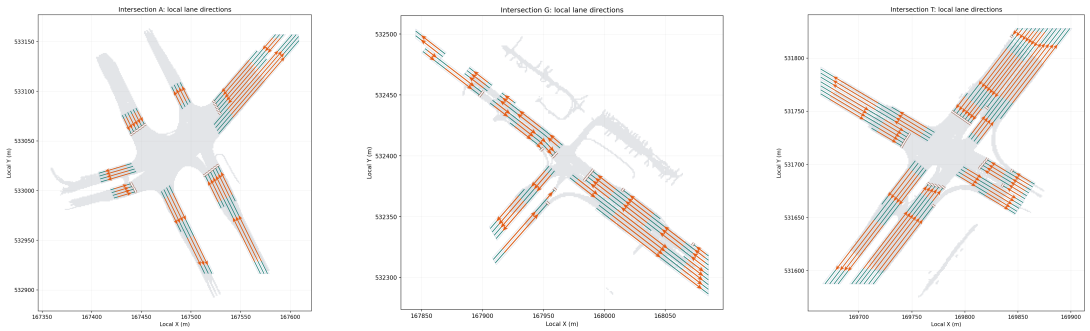


Figure 2: Overall composition of filtered segments across all exclusion categories.

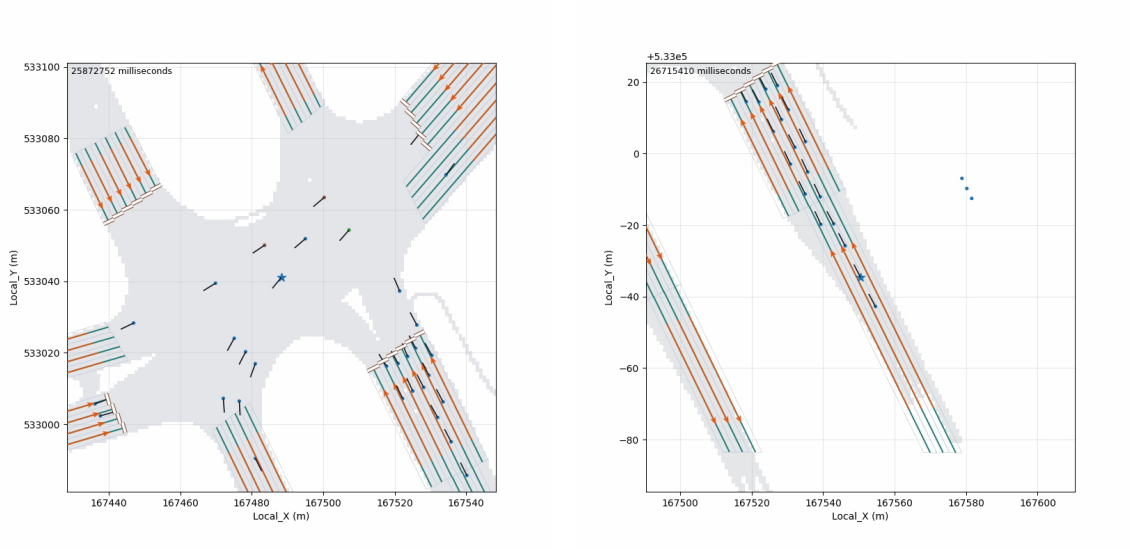
The Songdo Drive dataset covers various intersection types with sophisticated lane layouts and diverse vehicle motion patterns, which are very challenging for motion forecasting methods. Figure 3 shows three representative intersections with the derived map information, including the lane

directions, stop lines, and drivable area. All the intersections have traffic from multiple directions, and they have mixed lane types, including merging and splitting lanes, which introduce complicated cross-vehicle interactions and multimodal behavior. Figure 4 shows two snapshots of vehicles traversing an intersection and joining a queue, where the ego vehicle is surrounded by multiple vehicles with different travel destinations.



(a) Intersection A (b) Intersection G (c) Intersection T

Figure 3: Examples of processed lane-direction maps for three intersections in the proposed drone-based dataset.



(a) Traversing the intersection (b) Joining a queue

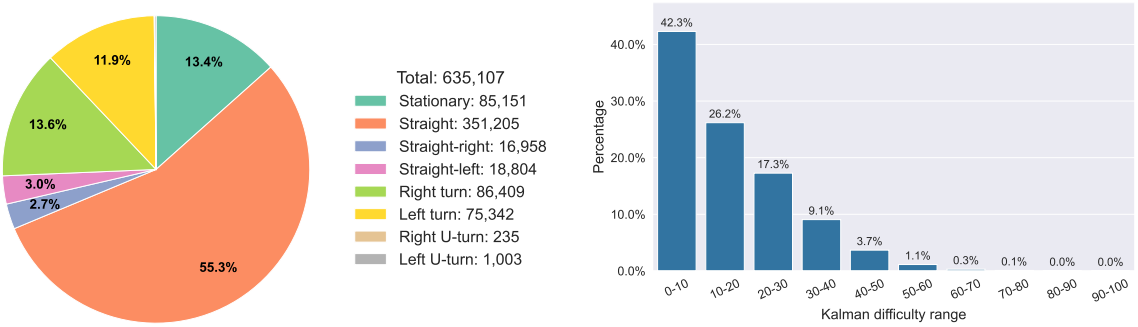
Figure 4: Example snapshots at intersection A.

Figure 5a shows the distribution of the trajectory types in the training split, where the type definitions follow the taxonomy in UniTraj (Feng et al., 2024). The movement patterns are categorized into eight types as shown in the figure, and all of them have reasonable presence in the dataset, with the straight motion being the most common type.

Aside from trajectory types, the variety of motion patterns can also be reflected by the Kalman difficulty score in Figure 5b, which is the FDE between the ground truth and a linear Kalman filter prediction (Feng et al., 2024). The distribution demonstrates that most trajectories are relatively easy to predict, and that motivates the evaluation by Kalman difficulty levels.

Motion Forecasting Experiments

In this section, we demonstrate the transferability challenge of motion forecasting models in a new city, and benchmark the performance of three representative methods, i.e., AutoBot (Girgis et al., 2022), MTR (Shi et al., 2022), and Wayformer (Nayakanti et al., 2023), on the Songdo Drive dataset. We utilized the implementation in UniTraj and kept the default configurations for all



(a) Trajectory type distribution in train split

(b) Kalman-based difficulty distribution

Figure 5: Training-split distributions of trajectory types and Kalman-based difficulty.

methods. Table 2 presents the overall evaluation metrics for these methods along with the training consumption on an H100 GPU. MTR* denotes an MTR model trained on the nuScenes dataset and evaluated directly on Songdo Drive Caesar et al. (2020), which is a zero-shot scenario that effectively demonstrates the challenge of domain shift.

Among the models trained on Songdo Drive, Wayformer achieves the best performance across all metrics, while AutoBot is a close second with lower training time requirements. Notably, the MTR model is a larger architecture with higher per-epoch training time, but it still delivers slightly worse performance than the lightweight AutoBot. Compared to the trained MTR model, the zero-shot MTR* has 108% higher BrierFDE and 118% higher minADE. This severe degradation confirms the pronounced domain shift challenge and the effectiveness of drone-based trajectory data for adapting the motion forecasting models.

Table 2: Evaluation results on the proposed Songdo Drive Dataset. MTR* denotes the off-the-shelf MTR model trained on nuScenes dataset, while the others are trained and tested on Songdo Drive. Best values are shown in bold.

Method	Epochs	Time (h)	BrierFDE	minADE	minFDE	Miss Rate
MTR*	–	–	3.953	1.497	3.369	0.494
AutoBot	100	18.713	1.847	0.588	1.197	0.154
Wayformer	150	22.864	**1.763**	**0.564**	**1.157**	**0.148**
MTR	40	32.969	1.902	0.680	1.516	0.273

Table 3: BrierFDE by trajectory type for the compared methods. Best values in bold.

Method	Stationary	Straight	Straight-right	Straight-left	Right turn	Left turn	Right U-turn	Left U-turn
MTR*	2.017	2.704	5.337	4.549	7.825	7.131	6.546	12.252
AutoBot	0.623	1.791	2.700	2.423	2.459	2.396	6.222	**4.513**
Wayformer	0.562	**1.690**	**2.631**	**2.262**	**2.399**	**2.350**	**4.857**	4.951
MTR	**0.457**	1.817	2.985	2.668	2.667	2.554	5.993	5.024

The per-type breakdown in Table 3 further reveals that stationary and straight movements are noticeably easier to cope with, whereas turning maneuvers and especially U-turns are much more challenging. Wayformer achieves consistent best BrierFDE on most trajectory types, demonstrating the most robust behavior across diverse motion patterns. AutoBot performs similarly on common motions, with a gap to Wayformer at $0.1 \sim 0.2$, and yields the best result on left U-turns. MTR is best on stationary trajectories despite its weaker overall performance. These results suggest that the variation in trajectory types in Songdo Drive provides a comprehensive benchmark for model performance, and complex maneuvers such as U-turns can particularly distinguish the capability of different models. In more complicated motion scenarios, the zero-shot and trained MTR models have a large performance gap, which again highlights the value of drone-based data for transferability.

A similar trend has been observed in the evaluation by Kalman difficulty in Table 4. The errors

Table 4: BrierFDE by Kalman difficulty for the compared methods. Best values in bold.

Method	Easy [0, 30)	Medium [30, 50)	Hard [50, 100]
MTR*	3.248	8.088	16.975
AutoBot	1.722	2.600	3.564
Wayformer	**1.645**	**2.482**	**3.242**
MTR	1.765	2.716	4.107

of all methods increase from easy to hard cases, confirming that the difficulty score is well aligned with forecasting complexity in Songdo Drive. Wayformer remains the best method across all three difficulty levels, with a more obvious advantage on the hard subset. Meanwhile, compared to zero-shot inference, the trained MTR model has a 75.8% improvement on the hard cases, which once more emphasizes the effectiveness of using drone-based trajectories for adapting motion forecasting models.

Table 5: Cross-preprocessing evaluation of AutoBot. The model is trained with the preprocessing scheme from one of the three models, and evaluated with the scheme from another. Numbers in parentheses indicate the number of surrounding vehicles.

	← **Evaluation** →			
↓ **Training**	**AutoBot (15)**	**Wayformer (32)**	**MTR (64)**	**Average**
AutoBot (15)	**1.847**	**1.811**	2.827	**2.162**
Wayformer (32)	1.916	1.832	2.767	2.172
MTR (64)	3.461	3.271	**1.818**	2.850

A motion forecasting model often relies on a model-specific preprocessing step to convert the raw scene data into an acceptable format, which includes selecting a number of surrounding vehicles and defining the map representation. Specifically, AutoBot, Wayformer, and MTR use 15, 32, and 64 surrounding vehicles respectively, and MTR has a different map representation than the other two. To evaluate the impact of preprocessing choices, we train and evaluate the AutoBot model across the three preprocessing schemes that were originally adopted by the three different models, and the results are shown in Table 5.

Despite the difference in the number of surrounding vehicles, the preprocessing schemes of AutoBot and Wayformer are quite similar. In general, when the training and testing preprocessing are consistent, the AutoBot model delivers consistent performance with BrierFDE around 1.8. The model trained with more surrounding agents (Wayformer preprocessing) has slightly worse performance when tested with fewer background vehicles (AutoBot preprocessing), but the gap is not significant. On the other hand, the different map representation of MTR places it in a different category from the other two. The model trained with MTR preprocessing has much worse performance when evaluated with AutoBot and Wayformer preprocessing, and vice versa, which suggests that the map representation is a more critical factor than surrounding vehicles in this case.

4 Conclusions

This work proposes an alternative perspective of transferring motion forecasting models to new cities via drone-based traffic monitoring. We show that diverse and large-scale driving scenes can be efficiently created from the broad coverage of drone observations. Experiments have demonstrated that an off-the-shelf motion forecasting model struggles to generalize to a new city due to heavy domain shift, while models trained on the drone-derived data can achieve much better performance. We hope our findings can inspire further research on the generalization of motion forecasting models with drone footage, and that our proposed benchmark can serve as an interesting resource for the community. In the future, we plan to explore the possibility of predicting the joint future of multiple traffic participants, and investigate the potential of using drone-based traffic monitoring to learn data-driven traffic simulation models. Another direction will be to enhance the rule compliance of

data-driven models by reinforcement learning.

Acknowledgements

This work is supported by the Swiss National Science Foundation (SNSF) under NCCR Automation, grant agreement number 51NF40_180545. We are grateful to Dr. Robert Fonod and Lan Feng for insightful discussions and support for this work.

References

Barmpounakis, E., & Geroliminis, N. (2020). On the new era of urban traffic monitoring with massive drone data: The pneuma large-scale field experiment. *Transportation research part C: emerging technologies*, *111*, 50–71.

Caesar, H., Bankiti, V., Lang, A. H., Vora, S., Liong, V. E., Xu, Q., ... Beijbom, O. (2020). nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 11621–11631).

Ettinger, S., Cheng, S., Caine, B., Liu, C., Zhao, H., Pradhan, S., ... others (2021). Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset. In *Proceedings of the ieee/cvf international conference on computer vision* (pp. 9710–9719).

Feng, L., Bahari, M., Amor, K. M. B., Zablocki, É., Cord, M., & Alahi, A. (2024). Unitraj: A unified framework for scalable vehicle trajectory prediction. *arXiv preprint arXiv:2403.15098*.

Fonod, R., Cho, H., Yeo, H., & Geroliminis, N. (2025). Advanced computer vision for extracting georeferenced vehicle trajectories from drone imagery. *Transportation Research Part C: Emerging Technologies*, *178*, 105205.

Girgis, R., Golemo, F., Codevilla, F., Weiss, M., D'Souza, J., Ebrahimi Kahou, S., ... Pal, C. J. (2022). Latent variable sequential set transformers for joint multi-agent motion prediction. In *International conference on learning representations (iclr)*.

Li, Q., Peng, Z. M., Feng, L., Liu, Z., Duan, C., Mo, W., & Zhou, B. (2023). Scenarionet: Open-source platform for large-scale traffic scenario simulation and modeling. *Advances in neural information processing systems*, *36*, 3894–3920.

Nayakanti, N., Al-Rfou, R., Zhou, A., Goel, K., Refaat, K. S., & Sapp, B. (2023). Wayformer: Motion forecasting via simple & efficient attention networks. In *2023 ieee international conference on robotics and automation (icra)* (pp. 2980–2987).

Shi, S., Jiang, L., Dai, D., & Schiele, B. (2022). Motion transformer with global intention localization and local movement refinement. *Advances in Neural Information Processing Systems*, *35*, 6531–6543.

Zhou, K., Liu, Z., Qiao, Y., Xiang, T., & Loy, C. C. (2023). Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *45*(4), 4396-4415. doi: 10.1109/TPAMI.2022.3195549