IEEE Robotics and Automation Letters (RAL) paper presented at
2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

# Multi-class Trajectory Prediction in Urban Traffic using the View-of-Delft Prediction Dataset

Hidde J-H. Boekema, Bruno K.W. Martens, Julian F.P. Kooij, and Dariu M. Gavrila

***Abstract*—This paper presents View-of-Delft Prediction, a new dataset for trajectory prediction, to address the lack of on-board trajectory datasets in urban mixed-traffic environments. View-of-Delft Prediction builds on the recently released urban View-of-Delft (VoD) dataset to make it suitable for trajectory prediction. Unique features of this dataset are the challenging road layouts of Delft, with many narrow roads and bridges, and the close proximity between vehicles and Vulnerable Road Users (VRUs). It contains a large proportion of VRUs, with 569 prediction instances for vehicles, 347 for cyclists, and 934 for pedestrians. We additionally provide high-definition map annotations for the VoD dataset to enable state-of-the-art prediction models to be used.**

**We analyse two state-of-the-art trajectory prediction models, PGP and P2T, which originally were developed for vehicle-dominated traffic scenarios, to assess the strengths and weaknesses of current modelling approaches in mixed traffic settings with large numbers of VRUs. Our analysis shows that there is a significant domain gap between the vehicle-dominated nuScenes and VRU-dominated VoD Prediction datasets. The dataset is publicly released for non-commercial research purposes.**

***Index Terms*—Datasets for Human Motion; Data Sets for Robot Learning; Deep Learning Methods**

## I. Introduction

THE ability to accurately forecast the trajectories of nearby traffic agents is key to automated driving as it enables planning a comfortable, collision-free path for an automated vehicle (AV). While this research topic has gained much attention, the bulk of prediction datasets recorded on-board a vehicle involve suburban areas [1] or cities with limited interaction between different road user classes [2], [3], [4], [5]. Urban settings with mixed traffic - where road infrastructure is shared between multiple different classes of agents - are comparatively little studied, despite these settings being especially challenging for prediction due to their unique road layouts, complex interactions between road users, and close proximity of vehicles to Vulnerable Road Users (VRUs) such as pedestrians and cyclists. Furthermore, forecasting the motion of VRUs in mixed traffic is difficult as they are highly manoeuvrable and can change their behaviour seemingly unpredictably.

Manuscript received: November, 8, 2024; Revised February, 11, 2024; Accepted March, 13, 2024.

This paper was recommended for publication by Editor A. Faust upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by the Dutch Research Council (NWO), within the Efficient Deep Learning (EDL) project (project number: P16-25).

The authors are with the Intelligent Vehicles Group, TU Delft, 2628 CD Delft, Netherlands (e-mail: h.j.boekema@tudelft.nl).

Digital Object Identifier (DOI): see top of this page.



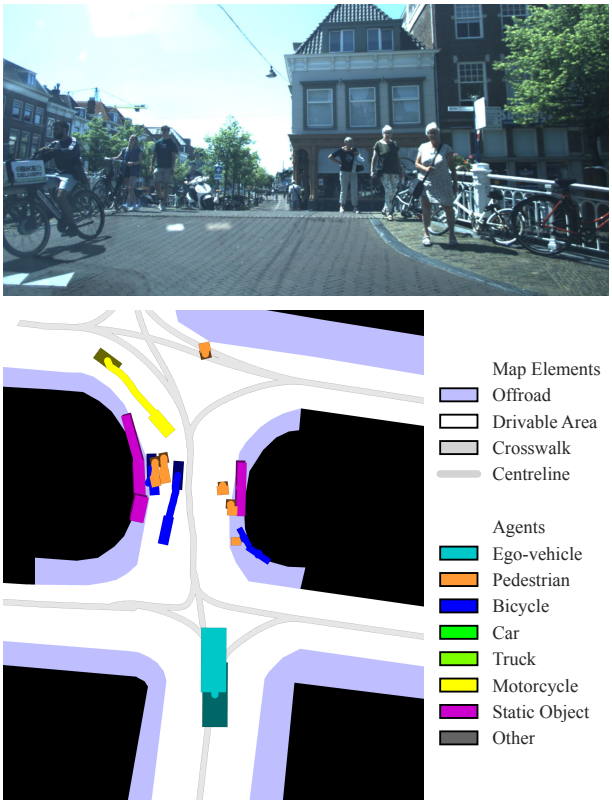



Fig. 1: Camera and topdown view of a scenario in the VoD Prediction dataset. The annotated semantic map is visualised in the rasterised topdown view using the nuScenes plotting code. Shaded agent boxes indicate past observations.

Trajectory prediction approaches for vehicles have been actively researched in recent years; see surveys [6], [7], [8], [9]. Typical deep learning-based approaches for this task use an encoder-decoder architecture e.g. [10], [11], [12]. The encoder transforms the past trajectory of agents, their interactions, and the local map information into a feature representation of the scene. The decoder predicts possible future trajectories for the agents from such representations. Current state-of-the-art models use 'vectorised' inputs to encode salient information for the prediction task [10], [13], [14]. This type of input efficiently represents road elements from map data and motion states of tracked agents as vectors, which avoids issues associated with rasterised representations of the environment e.g. lossy rendering and computational inefficiency [13], [14]. Vectorised semantic map information has been used with great success to learn the influence of the static environment on a vehicle's

trajectory. This approach not only improves scene compliance but also the accuracy of predictions for vehicles [10], [14], [15].

Despite these developments, predicting the trajectories of VRUs remains challenging. Most existing datasets focus on vehicle prediction [1], [2], [3], [4], [5]. We therefore present the View-of-Delft (VoD) Prediction dataset, an urban prediction dataset set in the historic city centre of Delft, the Netherlands, with challenging attributes such as unique road layouts, including many narrow roads and bridges, and close proximity between vehicles and VRUs. This dataset is an extension of the VoD dataset [16], and adds semantic map data, which state-of-the-art trajectory prediction methods depend on for accurate prediction. An example camera image from the VoD dataset and the corresponding prediction scenario are shown in Figure 1. To investigate to what degree trajectory prediction methods are well suited to challenging VRU-dominated environments, we evaluate *Prediction via Graph-based Policy* (PGP) [10], a state-of-the-art graph-based trajectory prediction approach, *Plans-to-Trajectories* (P2T), a raster-based approach (to compare this paradigm to graph-based methods), and, as a third common baseline, the constant velocity Kalman Filter (KF) on our novel VoD Prediction dataset.

## II. Related Work

### A. Trajectory Prediction

A considerable amount of literature exists on trajectory prediction of traffic agents in the context of automated driving; see [6], [7], [8], [9] for surveys on this topic. However, the majority of current work focuses on predicting the future trajectory for a single class of road user, e.g. cars [10], [17], [18], cyclists [19], or pedestrians [20], [21], [22]. Few approaches are designed to forecast the future trajectories of multiple classes of agents [17], [23], [24].

Trajectory prediction approaches mainly use static scene information in one of two formats: as raster maps, which render geometric information as an image, or vector maps, which provide this information as geometric primitives such as points, polylines, and planes. An example of a raster-based approach is Plans-to-Trajectories (P2T) [25], which conditions trajectory predictions on plans from a grid-based policy learnt using inverse reinforcement learning (IRL). In recent years, vectorised representations [10], [13], [26] have gained popularity as a means to encode scene information, as they do not suffer from the loss in spatial and semantic information inherent to rasterised representations [13]. Prediction via Graph-based Policy (PGP) [10] effectively leverages this representation to sample feasible trajectories for vehicles over the lane graph. This is an advantage over popular goal-conditioned prediction methods [17], [18], [27], [28], which only take the feasibility of the selected goal location into account, and not the feasibility of the possible routes to a goal as well [15]. A loss function that uses vector map data to enforce driving rules for vehicles is proposed in [29]. Finally, Graph Neural Networks (GNNs) are a natural choice for modelling the interactions between traffic agents and the road topology with this representation, and have been shown to improve prediction performance [26].

### B. Motion Prediction Datasets

Numerous trajectory prediction datasets have been released in the past few years [1], [2], [3], [4], [5], [11]. We limit our discussion to datasets that were recorded from a moving vehicle ("on-board" setting) as this setting is the most relevant to automated driving systems. Desirable features for datasets intended for trajectory prediction in mixed traffic include: 1) contain ground truth tracks of multiple agent classes, 2) have a high number of interactions between different agent classes, 3) contain semantic map annotations, and 4) provide sensor data to enable the use of subtle context cues. In Table I we show the compliance of commonly-used datasets with most of these features.

Many of the current datasets for on-board trajectory prediction either have a significantly greater number of vehicle annotations than any other road user class or are comprised of highly structured traffic scenarios in suburban or regional locations and thus have little interactions between agent classes. This limits the transferability of models developed on these datasets to urban mixed-traffic settings. Some of these datasets, such as nuScenes [2] and Argoverse 1 [3], do not even have any pedestrian or cyclist annotations for prediction in their test set [11], and are hence not suitable for multi-class trajectory prediction.

Accurate and detailed semantic map annotations have become crucial due to the increasing use of map features by prediction models [4], [29]. However, not all trajectory prediction datasets provide map annotations. An example of a dataset without map annotations is Euro-PVI [11], one of the largest urban European datasets for trajectory prediction. Amongst the datasets that do provide map annotations, there is variability in the quality of the annotations. For example, the Argoverse 1 Prediction [3] maps only contain lane centerline and drivable area annotations, whereas Argoverse 2 [4] and nuScenes [2] provide more information e.g. lane boundaries, road signs, traffic lights. The Argoverse 2 and Waymo Motion [5] annotations are 3D, in contrast to nuScenes.

Sensor data can be used to develop more effective prediction frameworks as they provide supplemental information to agent tracks. However, most large-scale trajectory datasets either do not provide sensor data such as camera images, LiDAR point clouds, or radar data (e.g. Lyft Level 5 Prediction [1], Argoverse 1 and 2) or release it in a limited form. For example, the 1.2.0 release of Waymo Motion adds LiDAR data for the first second of agent tracks only.

### C. Contributions

Our contributions are twofold:

1) We release the naturalistic VoD Prediction dataset[1], an extension of the urban VoD dataset [16]. This dataset contains a large proportion of VRUs such as pedestrians and cyclists. It additionally has high-quality 3D road user annotations, vectorised semantic map elements such as lanes, crosswalks, intersections and off-road areas, plus sensor data from camera, LiDAR, and radar. The

[1]https://intelligent-vehicles.org/datasets/view-of-delft

TABLE I: Overview of on-board motion prediction datasets, with their released sensor information, semantic map information, size of the dataset, number of agents (and agent classes) for prediction, and recording locations. VoD Prediction is our dataset. N.A. = North America. * All agents reported because breakdown of agents intended for prediction is not reported.

| Dataset | Location | Information | | | | Size | | | Agents for Prediction | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | Camera | LiDAR | Radar | Sem. Map | Scenes | Duration (s) (hist., fut.) | Freq. (Hz) | Vehicles | Cyclists | Pedestrians | # Pred. Classes |
| Lyft Level 5 [1] | N.A. | ☐ | ☐ | ☐ | ☑ | 170k | 0.5, 5 | 10 | *49M | *77k | *605k | 3 |
| WOMD [5] | N.A. | ☐ | ☐ | ☐ | ☑ | 104k | 1, 8 | 10 | *60k | *620 | *23k | 3 |
| Argoverse 2 [4] | N.A. | ☐ | ☐ | ☐ | ☑ | 250k | 5, 6 | 10 | 10k | 1000 | 1000 | 5 |
| nuScenes [2] | N.A., Asia | ☑ | ☑ | ☑ | ☑ | 1000 | 2, 6 | 2 | 1000 | 0 | 0 | 1 |
| Euro-PVI [11] | Europe | ☑ | ☑ | ☐ | ☐ | 1077 | 0.5, 3 | 2 | 1077 | 1581 | 6177 | 3 |
| **VoD Prediction** | **Europe** | ☑ | ☑ | ☑ | ☑ | **1850** | **0.5, 3** | **10** | **569** | **347** | **934** | **3** |

dataset is accompanied by a software kit[2] to facilitate using the dataset for customized motion prediction scenarios. It is additionally available in nuScenes format.

2) We investigate how well two representative state-of-the-art trajectory prediction approaches (PGP [10] and P2T [25]) predict in mixed-traffic urban settings. We also investigate the domain gap between the vehicle-dominated nuScenes and urban View-of-Delft Prediction datasets using the PGP model.

## III. Trajectory Prediction

The aim of trajectory prediction is to estimate the future trajectory of one or more 'target' agent(s) over a time horizon $T_f$ given an observed history of length $T_h$. The observed states of each agent $a$ in a prediction scenario are given as a sequence $\mathbf{x}^a_{-T_h:0} = \{\mathbf{x}^a_{-T_h}, ..., \mathbf{x}^a_{-1}, \mathbf{x}^a_0\}$, where each state $\mathbf{x}^a_t$ contains the pose and velocity of the agent at time $t$. Map data, semantic attributes (e.g. agent class) and/or sensor data can be used as additional context information. Given these inputs, a prediction model may make $K$ predictions for the future trajectory of the target agent. The predictions are evaluated against the ground truth future trajectory $\mathbf{y}^a_{1:T_f} = \{\mathbf{y}^a_1, ..., \mathbf{y}^a_{T_f-1}, \mathbf{y}^a_{T_f}\}$, where each $\mathbf{y}^a_t$ is the position of the agent at time $t$. In multi-class trajectory prediction, target agents can be from different classes and/or share the environment with other road user classes.

## IV. Dataset

In this section, we present the View-of-Delft Prediction (VoD-P) dataset, an extension of the View-of-Delft (VoD) dataset [16] for trajectory prediction. The VoD dataset comprises camera, radar, LiDAR, and GPS/IMU information. Tracked objects were annotated at 10 Hz, in contrast to e.g. nuScenes, which was annotated at only 2 Hz. A summary of the dataset and comparison with major trajectory prediction datasets can be found in Table I. For more details on the sensor setup, we refer readers to [16]. Here, we outline the specific additions that allow the dataset to be used for trajectory prediction for vehicles and VRUs.

[2]https://github.com/tudelft-iv/vod-devkit

### A. Scene Selection

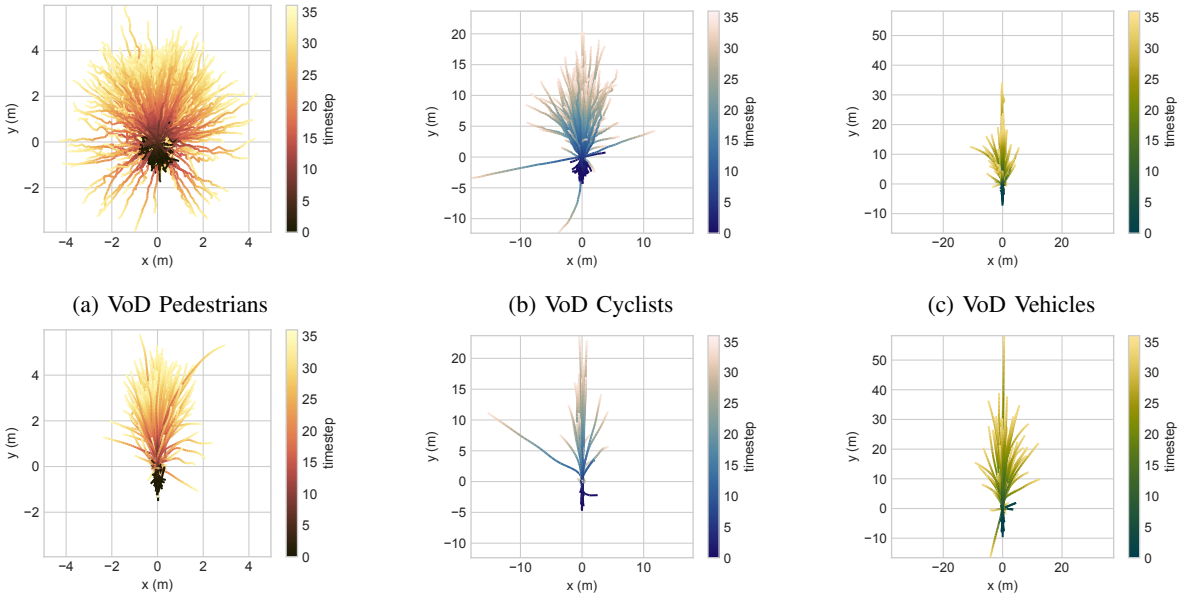
The View-of-Delft dataset is recorded in locations with a large number of VRUs and a high degree of interaction between different road user classes, such as pedestrians, cyclists, and cars, amongst others. We use this data to create prediction scenes consisting of tracks and context over a 'history' and 'future' period for a single 'target' agent. Agents are selected as prediction targets if they have been observed for at least the entire history and future period. Thus we have full trajectory information over both the history and future for each prediction target, contrary to datasets that allow target agents to have partially observed histories e.g. nuScenes [2]. We do take agents that are partially observed into account as surrounding agents. Parked vehicles and bicycles without riders are not included as target agents.

We modify the original VoD [16] dataset splits to better suit the prediction task. The train and validation split are merged into one train-val split as in nuScenes to allow more efficient use of the data available. No target agents are taken from VoD highway sequence as the scope of the VoD-P dataset is urban driving (the highway sequence, including map annotations, is still provided in VoD-P). Target agents from the same recording are assigned to the same split. This results in a 76%/24% train-validation/test split for an $0.5$ s history and 3 s prediction horizon. Note that only recordings from the VoD test split are in the VoD-P test split, and similarly for the combined train and validation splits. See Figure 2 for the distribution of tracks of target agents in the train and validation set; VoD-P has a significantly larger spread in pedestrian and cyclist movement directions than nuScenes [2].

Table II shows quantitative dataset statistics. This table shows the distribution of agents in the dataset, as well as trajectory statistics by agent class: the mean speed and acceleration, max acceleration, greatest in-trajectory speed difference, and path efficiency (see [30] for a definition). See also [30] for the statistics of other prediction datasets.

TABLE II: VoD-P dataset statistics.

| Agent Class | Total Time (h) | % of Agents | | Speed $(m/s)$ | | Accel. $(m/s^2)$ | | Path Eff. (%) |
|---|---|---|---|---|---|---|---|---|
| | | Train | Test | Mean | Diff. | Mean | Max | |
| Pedestrian | 0.91 | 50.1 | 51.8 | 0.82 | 0.37 | -0.03 | 2.53 | 90.2 |
| Cyclist | 0.34 | 18.7 | 18.9 | 3.19 | 0.80 | -0.06 | 2.28 | 97.8 |
| Vehicle | 0.55 | 31.2 | 29.3 | 2.56 | 0.86 | -0.02 | 7.16 | 93.5 |




(a) VoD Pedestrians (b) VoD Cyclists (c) VoD Vehicles

(d) nuScenes Pedestrians (e) nuScenes Cyclists (f) nuScenes Vehicles

Fig. 2: Samples of VoD-P and nuScenes tracklets in the training and validation sets for a history of $T_h = 0.5$ s and future of $T_f = 3$ s. An equal number of samples was randomly drawn from the datasets to ensure a fair comparison. The tracklets are positioned at the origin and aligned with the positive y-axis at $t = 0$ (i.e. last observed timestep). The VoD-P dataset contains challenging and diverse tracklets.

### B. Vector Map Information

As current trajectory prediction methods [10], [17], [18], [25] rely on semantic map data, we provide accurate annotations of designated lanes, intersections, crosswalks and off-road areas with extensive labels for each road element. This means that our dataset contains information that can aid the prediction performance for agents in urban areas. The annotations were created by human annotators from georeferenced aerial images[3]. Additionally, we label every road element with a unique element identifier, its road type and which road users are allowed to use the road element.

**Lanes** are annotated as polygons, denoting the drivable area of the lane. The boundaries and lane centreline are calculated from these polygons; their direction denotes the legal direction of travel along the lane. We also provide the following labels: road type (e.g. residential, bike path, highway), the agent classes allowed to use the lane, and the type of road boundary (i.e. solid/dashed marking). Road boundary types determine which lane switches are feasible. **Intersections** are annotated as a polygon that encloses the area of an intersection. We define an intersection as a region where at least one lane ends *and* at least one lane starts. By this definition, lane merges or splits also occur on an intersection. Lanes that agents are legally allowed to travel between are annotated as connected. A "connecting" lane between them is calculated by interpolating the boundaries and centrelines of the lanes over the intersection. **Crosswalks** are polygons that indicate designated pedestrian crossing locations. **Off-road areas** indicate the pedestrian domain, e.g. sidewalks, city squares, footbridges, as well as other non-road areas such as car parks. These polygons can help to predict for pedestrians.

[3]https://www.pdok.nl/wms

### C. Vehicle Localisation

In order to ensure that the tracks in the dataset are smooth and accurately positioned with respect to the map data, we estimate the 6-DOF pose of the ego-vehicle at each timestamp using a two-stage localisation pipeline. First, we apply KISS-ICP [31] on the LiDAR point clouds to get estimates of the relative transformation between consecutive poses of the ego-vehicle. Global pose estimates are obtained by manual annotation of point correspondences in the camera images and world frame of selected timestamps. The poses are calculated from the correspondences using Perspective-n-Point (PnP) [32]. Finally, we fuse the local and global transforms using Pose Graph Optimisation (PGO) [33].

## V. Methodology

We analyse the performance of state-of-the-art trajectory prediction approaches developed on vehicle-dominated datasets on the VRU-dominated View-of-Delft Prediction dataset. As baseline for this analysis, we select PGP [10] because it is state-of-the-art for the vehicle-heavy nuScenes dataset [2]. We also select the raster-based P2T [25] model as a baseline to investigate the difference between the graph-based and raster-based map encoding paradigms for complex map data. We summarize the PGP and P2T architectures, which are also illustrated in Figure 3.

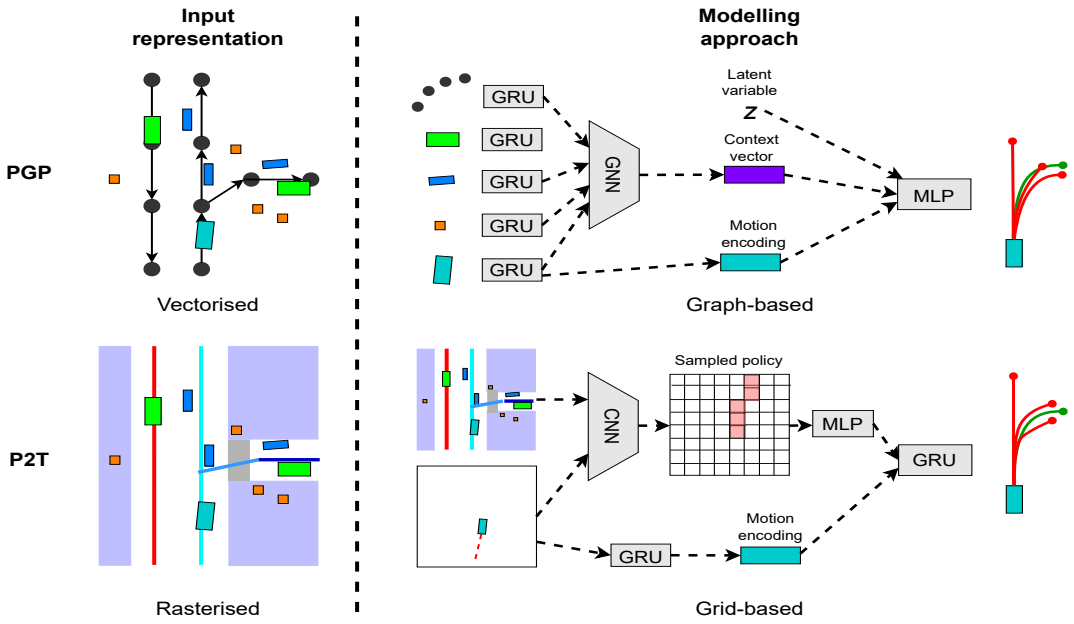



Fig. 3: Summary of PGP [10] and P2T [25] models. PGP uses an architecture based on Graph Neural Networks (GNN) to encode a vectorised representation of the scene. P2T estimates and fine-tunes discrete policies from a rasterised representation using a Convolutional Neural Network (CNN).

*a) PGP:* PGP [10] consists of three modules: a graph encoder, a policy header and a trajectory decoder. The graph encoder encodes vectorised lane information using a Graph Neural Network (GNN) consisting of graph attention layers [34]. The motion (relative to the pose of the target agent at $t = 0$) and class, represented as one-hot vector (one element has value 1 and all others 0), of the target agent and surrounding agents is encoded using Gated Recurrent Units (GRUs) [35]. Social interactions are encoded using multi-head attention and are represented as node features on the constructed graph. Next, the policy header uses the motion encoding and graph encodings to estimate the transition probability between connected nodes in the lane graph. Samples are drawn from this policy serve as social and scene context for the trajectory decoder. These samples are combined with the encoded motion of the target agent and a sample from a latent noise distribution, and refined by a Multi-Layer Perceptron (MLP) to obtain the future trajectory predictions.

*b) P2T:* P2T [25] also consists of three modules: a convolutional reward model for path and goal states, a Maximum Entropy Inverse Reinforcement Learning (MaxEnt IRL) [36] policy that can be sampled to obtain discrete plans on the 2D grid, and an attention-based trajectory decoder. Scene and social context is encoded as a raster image, where the colour channels are used to represent semantic information, e.g., lane direction and agent class. This context and the observed trajectory of the target agent are first used to calculate the likelihood of the agent's future path and goal location on a discrete grid of the environment using a CNN. These path and goal maps are then used to estimate a policy for the agent, which is sampled to obtain discrete plans over the grid of the environment. Finally, the trajectory decoder outputs a continuous-valued trajectory for each discrete plan. The observed trajectory of the agent is also input to the trajectory decoder.

## VI. Experiments

### A. Experimental Setup

*1) Datasets:* In our experiments, we use both the VoD-P dataset and the nuScenes [2] dataset. We evaluate our models on the VoD-P dataset with a history of $T_h = 0.5$ s and a future of $T_f = 3$ s, following [1], [11]. See Table I for some dataset statistics of VoD-P. To compare nuScenes to our dataset, we truncate the tracks in their prediction challenge split and interpolate and re-sample their tracks at 10 Hz to match the setup of VoD.

*2) Metrics:* We adopt the widely used minimum Average Displacement Error (minADE) (↓) and Miss Rate (MR) (↓) metrics for $K = \{5, 10\}$ predictions. The minADE is the lowest average Euclidean distance between the ground truth trajectory $\mathbf{y}$ and set of $K$ predicted trajectories $\{\hat{\mathbf{y}}^{(1)}, ..., \hat{\mathbf{y}}^{(K)}\}$ over the prediction horizon $T_f$ and a dataset of size $N$:

$$\min \text{ADE}_K = \frac{1}{N} \sum_{n=1}^{N} \min \text{ADE}_{K,n}, \qquad \text{where} \tag{1}$$

$$\min \text{ADE}_{K,n} = \min_{i \in \{1,...,K\}} \frac{1}{T_f} \sum_{t=1}^{T_f} \left\| \mathbf{y}_{t,n} - \hat{\mathbf{y}}_{t.n}^{(i)} \right\|_2 . \tag{2}$$

The MR is the fraction of scenes wherein the maximum pointwise L2 distance between a predicted track $\hat{\mathbf{y}}_{1:T}$ and the ground truth $\mathbf{y}_{1:T}$ is greater than a threshold distance $r$ for any of the $K$ predictions:

$$\text{MR}_K = \frac{1}{N} \sum_{n=1}^{N} \text{Miss}_{K,n}, \qquad \text{where} \tag{3}$$

TABLE III: Prediction performance of PGP [10] model for a history of $T_h = 0.5$ s and a prediction horizon of $T_f = 3$ s and various training configurations.

| Evaluation Dataset | Training Datasets | | K=1 | K=5 | | K=10 | |
|---|---|---|---|---|---|---|---|
| | nuScenes | VoD-P | minFDE ↓ | minADE ↓ | MR ↓ | minADE ↓ | MR ↓ |
| nuScenes [2] | ☑ | ☐ | 2.88 | 0.50 | 0.78 | 0.36 | 0.67 |
| | ☐ | ☑ | 15.21 | 7.00 | 0.93 | 6.55 | 0.91 |
| VoD-P | ☑ | ☐ | 35.94 | 15.20 | 0.98 | 13.07 | 0.98 |
| | ☐ | ☑ | 2.24 | 0.66 | 0.60 | 0.56 | 0.48 |
| | ☑ | ☑ | 2.19 | 0.49 | 0.67 | 0.38 | 0.52 |

$$\text{Miss}_{K,n} = H\left(R_{K,n} - r\right), \qquad \text{where} \tag{4}$$

$$R_{K,n} = \min_{i \in \{1,\dots,K\}} \left( \max_{t \in \{1,\dots,T_f\}} \left\| \mathbf{y}_{t,n} - \hat{\mathbf{y}}_{t,n}^{(i)} \right\|_2 \right). \tag{5}$$

where $H(\cdot)$ is the Heaviside step function. We choose a threshold $r$ of 0.5 m for a prediction horizon of $T = 3$ s, to account for the distance agents can travel in this time.

*3) Baselines:* To assess the degree of linearity (and thereby the difficulty) of the dataset, we use a Kalman Filter with a linear dynamics model as a baseline. We smoothed the agent tracks in the training set by fitting a spline to the tracks and calculated the difference between the original and smoothed track points to estimate the measurement noise parameters. The process noise was estimated using the deviation of poses and velocities from the linear model in the smoothed tracks. Our deep learning baselines are the PGP [10] and P2T [25] models, described in Section V.

### B. Domain Gap Between VoD-P and nuScenes

We first investigate the domain gap between the urban VoD-P dataset and the vehicle-heavy nuScenes dataset by testing the PGP model on the VoD-P test set with different training setups: 1) training only on nuScenes, 2) training only on VoD-P, and 3) pre-training on nuScenes and fine-tuning on VoD-P. For a fair comparison between the datasets we modify the nuScenes tracklets to match the prediction setup of VoD-P ($T_h = 0.5$ s, $T_f = 3$ s @ 10 Hz). Table III shows the performance of the model for these setups.

Only training on nuScenes leads to poor prediction performance on the VoD-P test set; the minADE for $K = 10$ samples is $2234\%$ higher than when only training on VoD-P. This shows that there is a significant domain gap between the two datasets. Pre-training the model on nuScenes and fine-tuning on VoD-P gives better average minADE scores ($32\%$ lower for $K = 10$) than training on VoD-P alone, suggesting that the datasets are complementary and can be used together for developing better models for urban trajectory prediction. The high minFDE for $K = 1$ (over 2 m in all cases) shows that prediction models have to make significant progress to ensure safe and comfortable autonomous driving.

We also show the performance of PGP when trained and evaluated on nuScenes. Notably, it scores better on the minADE and MR metrics than when trained and evaluated on VoD-P, demonstrating the difficulty of our dataset.

### C. Quantitative Results

Table IV shows the performance of the baselines over 6 different train-test splits. Positional errors (minADE) tend to increase when considering pedestrians, cyclists and vehicles successively, due to the increased travelling speeds involved. Overall, PGP performs best. Interestingly, KF edges out the second spot. It does relatively well for vehicles, which mostly drive straight within their lanes, a behavior well captured by a constant velocity motion model with added noise. For the pedestrians and cyclists with greater freedom of movement, the KF is clearly inferior to PGP. For completeness, Table V shows the results of the baseline methods on the designated VoD-P train-test split, i.e. the one used for benchmarking (similar effects can be observed).

### D. Qualitative Results

Figure 4 shows the predictions of the baselines on an example scene for each target agent class. The examples show that the dynamics of agents in VoD-P are complex and non-linear.

The example illustrates how the P2T and PGP models are often able to account for the map context, but not always for the social context of the scene. The models are able to predict the turn that the car will make in Figures 4b and 4c, showing that they are able to process the dynamics of the vehicle and the road layout. Figures 4e and 4f show a failure case for a pedestrian for both models. The pedestrian in this scene walks perpendicular to the lane centrelines, and passes between a vehicle and pedestrian. PGP predicts that the pedestrian will walk into the vehicle blocking the road; P2T predicts wrongly that the pedestrian will avoid the obstacles. In Figures 4h and 4i a cyclists makes a turn to the left at the last minute. Only P2T accounts for this possibility in its predictions, but ignores the presence of another agent in its predictions. The failure cases may result from the high variance in VRU dynamics, or the varied and complex road layouts and social interactions between agents in urban centres. For example, cyclists and pedestrians can move orthogonal to or even against the driving direction of lanes. These examples show that the tested state-of-the-art models are unable to consistently predict the behaviour of agents in urban settings and suggests that predicting VRU trajectories might require a different treatment of map information than predicting vehicles bound to lanes.

## VII. Conclusions

We introduced the View-of-Delft Prediction (VoD-P) dataset, an extension of the VoD [16] dataset, enriching the available sensor data with vectorised map information. Our experiments show that there is a significant domain gap between the urban VoD-P dataset and the widely used nuScenes [2] dataset, highlighting the need for urban prediction datasets with many Vulnerable Road Users (VRUs). Our dataset is a step towards bridging the gap, enabling future research on trajectory prediction in complex urban traffic.

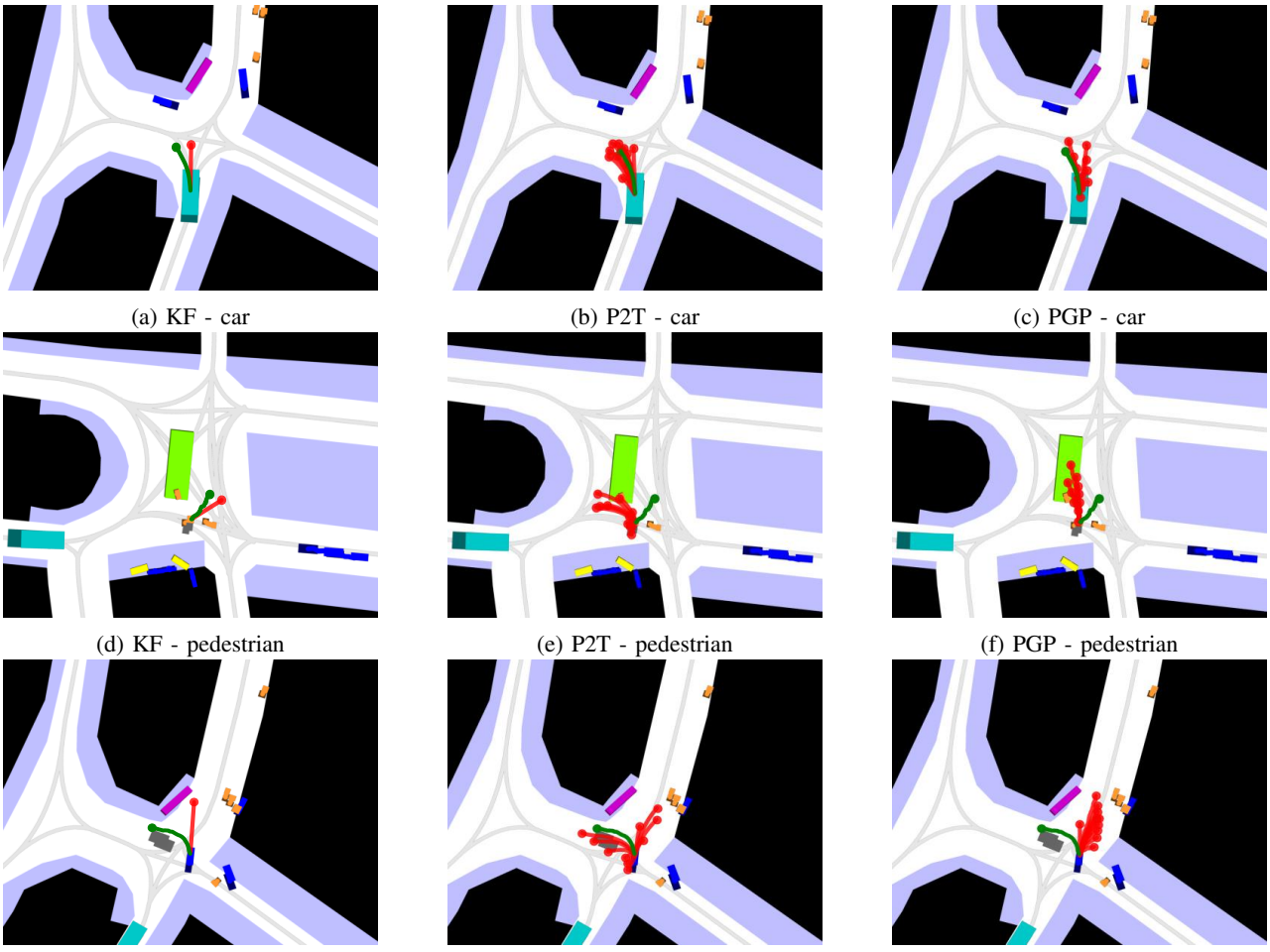
From the comparatively poor performance of a constant velocity Kalman Filter (KF) on VoD-P, we infer that there are a sizeable number of non-linear trajectory segments in the

TABLE IV: Performance over 6 different train-test splits of VoD-P with a $T_f = 3$ s prediction horizon for $K = 10$ samples. Mean $\pm$ std. dev. are over the test sets in the splits. Best performance on each metric is shown in **bold**. minADE = minimum Average Displacement Error, MR = Miss Rate.

| Method | Vehicle | | Cyclist | | Pedestrian | |
|---|---|---|---|---|---|---|
| | minADE ↓ | MR ↓ | minADE ↓ | MR ↓ | minADE ↓ | MR ↓ |
| KF | $0.81 \pm 0.04$ | $0.81 \pm 0.04$ | $1.25 \pm 0.06$ | $0.97 \pm 0.02$ | $0.52 \pm 0.03$ | $0.78 \pm 0.03$ |
| P2T [25] | $1.79 \pm 1.30$ | $0.78 \pm 0.07$ | $1.59 \pm 0.64$ | $0.93 \pm 0.04$ | $0.59 \pm 0.14$ | $0.59 \pm 0.13$ |
| PGP [10] | $\mathbf{0.48 \pm 0.30}$ | $\mathbf{0.39 \pm 0.08}$ | $\mathbf{0.50 \pm 0.07}$ | $\mathbf{0.74 \pm 0.04}$ | $\mathbf{0.29 \pm 0.03}$ | $\mathbf{0.41 \pm 0.06}$ |

TABLE V: Performance using the designated VoD-P train-test split with a $T_f = 3$ s prediction horizon for $K = 10$ samples. Mean $\pm$ std. dev. are over the target agents in the test set. All models were trained on VoD-P only. Best performance on each metric is shown in **bold**. minADE = minimum Average Displacement Error, MR = Miss Rate.

| Method | Vehicle | | Cyclist | | Pedestrian | |
|---|---|---|---|---|---|---|
| | minADE ↓ | MR ↓ | minADE ↓ | MR ↓ | minADE ↓ | MR ↓ |
| KF | $\mathbf{0.93 \pm 0.67}$ | 0.85 | $1.22 \pm 0.85$ | 0.99 | $0.52 \pm 0.38$ | 0.80 |
| P2T [25] | $2.24 \pm 3.38$ | 0.83 | $1.01 \pm 0.79$ | 0.95 | $0.42 \pm 0.31$ | 0.66 |
| PGP [10] | $1.09 \pm 1.88$ | **0.50** | $\mathbf{0.56 \pm 0.62}$ | **0.81** | $\mathbf{0.27 \pm 0.22}$ | **0.35** |



(a) KF - car (b) P2T - car (c) PGP - car

(d) KF - pedestrian (e) P2T - pedestrian (f) PGP - pedestrian

(g) KF - cyclist (h) P2T - cyclist (i) PGP - cyclist

Fig. 4: Qualitative examples of baseline model predictions on VoD-P dataset for $K = 10$ predictions. Predictions are shown in red; the ground truth future tracklet is shown in green. Refer to the legend in Figure 1 for the colour scheme of the map annotations and agents. In the top row scenario, PGP and P2T are able to predict the turn the car will make. However, in the second row scenario, PGP and P2T make inaccurate predictions because of the complexity of the scene. In the last row scenario, only P2T predicts the sharp turn of the cyclist. See also the videos in the Supplemental Material.

dataset. We also analysed the performance of the graph-based PGP [10] and raster-based P2T [25] models on our dataset. Although PGP outperformed P2T on the metrics, the results show that neither model is able to capture the full social and static context of prediction scenarios for VRUs as well as for vehicles, motivating the search for new prediction techniques in urban centres.

Future work includes encoding social and map context differently for each agent class to account for their unique interactions with other agents and the static environment. Furthermore, more experimentation on domain transfer is needed, how models transfer across regions and how regional datasets are best combined.

## ACKNOWLEDGEMENT

This work is part of the research programme Efficient Deep Learning (EDL) with project number P16-25, which is funded by the Dutch Research Council (NWO).

## REFERENCES

[1] J. Houston, G. Zuidhof, L. Bergamini, Y. Ye, L. Chen, A. Jain, S. Omari, V. Iglovikov, and P. Ondruska, "One thousand and one hours: Self-driving motion prediction dataset," in *Conf. on Rob. Learning*. PMLR, 2021, pp. 409–418.

[2] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuScenes: A multimodal dataset for autonomous driving," in *Proc. of the IEEE/CVF Conf. on CVPR*, 2020, pp. 11 621–11 631.

[3] M.-F. Chang, J. Lambert, P. Sangkloy, J. Singh, S. Bak, A. Hartnett, D. Wang, P. Carr, S. Lucey, D. Ramanan *et al.*, "Argoverse: 3d tracking and forecasting with rich maps," in *Proc. of the IEEE/CVF Conf. on CVPR*, 2019, pp. 8748–8757.

[4] B. Wilson, W. Qi, T. Agarwal, J. Lambert, J. Singh, S. Khandelwal, B. Pan, R. Kumar, A. Hartnett, J. K. Pontes *et al.*, "Argoverse 2: Next generation datasets for self-driving perception and forecasting," *arXiv preprint arXiv:2301.00493*, 2023.

[5] S. Ettinger, S. Cheng, B. Caine, C. Liu, H. Zhao, S. Pradhan, Y. Chai, B. Sapp, C. R. Qi, Y. Zhou *et al.*, "Large scale interactive motion forecasting for autonomous driving: The Waymo open motion dataset," in *Proc. of the IEEE/CVF Intl. Conf. on Comp. Vis.*, 2021, pp. 9710–9719.

[6] J. Liu, X. Mao, Y. Fang, D. Zhu, and M. Q.-H. Meng, "A survey on deep-learning approaches for vehicle trajectory prediction in autonomous driving," in *2021 IEEE Intl. Conf. on Rob. and Biomimetics (ROBIO)*. IEEE, 2021, pp. 978–985.

[7] A. Rudenko, L. Palmieri, M. Herman, K. M. Kitani, D. M. Gavrila, and K. O. Arras, "Human motion trajectory prediction: A survey," *The Intl. J. of Rob. Research*, vol. 39, no. 8, pp. 895–935, 2020.

[8] P. Karle, M. Geisslinger, J. Betz, and M. Lienkamp, "Scenario understanding and motion prediction for autonomous vehicles-review and comparison," *IEEE Trans. on Int. Transportation Systems*, 2022.

[9] S. Mozaffari, O. Y. Al-Jarrah, M. Dianati, P. Jennings, and A. Mouzakitis, "Deep learning-based vehicle behavior prediction for autonomous driving applications: A review," *IEEE Trans. on Int. Transportation Systems*, vol. 23, no. 1, pp. 33–47, 2020.

[10] N. Deo, E. Wolff, and O. Beijbom, "Multimodal trajectory prediction conditioned on lane-graph traversals," in *Conf. on Rob. Learning*. PMLR, 2022, pp. 203–212.

[11] A. Bhattacharyya, D. O. Reino, M. Fritz, and B. Schiele, "Euro-PVI: Pedestrian vehicle interactions in dense urban centers," in *Proc. of the IEEE/CVF Conf. on CVPR*, 2021, pp. 6408–6417.

[12] N. Lee, W. Choi, P. Vernaza, C. B. Choy, P. H. Torr, and M. Chandraker, "Desire: Distant future prediction in dynamic scenes with interacting agents," in *Proc. of the IEEE Conf. on CVPR*, 2017, pp. 336–345.

[13] J. Gao, C. Sun, H. Zhao, Y. Shen, D. Anguelov, C. Li, and C. Schmid, "VectorNet: Encoding hd maps and agent dynamics from vectorized representation," in *Proc. of the IEEE/CVF Conf. on CVPR*, 2020, pp. 11 525–11 533.

[14] M. Liang, B. Yang, R. Hu, Y. Chen, R. Liao, S. Feng, and R. Urtasun, "Learning lane graph representations for motion forecasting," in *Comp. Vis.–ECCV 2020: 16th European Conf., Glasgow, UK, August 23–28, 2020, Proc., Part II 16*. Springer, 2020, pp. 541–556.

[15] B. Kim, S. H. Park, S. Lee, E. Khoshimjonov, D. Kum, J. Kim, J. S. Kim, and J. W. Choi, "LaPred: Lane-aware prediction of multi-modal future trajectories of dynamic agents," in *Proc. of the IEEE/CVF Conf. on CVPR*, 2021, pp. 14 636–14 645.

[16] A. Palffy, E. Pool, S. Baratam, J. F. Kooij, and D. M. Gavrila, "Multi-class road user detection with 3+ 1d radar in the View-of-Delft dataset," *IEEE Rob. and Automation Letters*, vol. 7, no. 2, pp. 4961–4968, 2022.

[17] J. Gu, C. Sun, and H. Zhao, "DenseTNT: End-to-end trajectory prediction from dense goal sets," in *Proc. of the IEEE/CVF Intl. Conf. on Comp. Vis.*, 2021, pp. 15 303–15 312.

[18] H. Zhao, J. Gao, T. Lan, C. Sun, B. Sapp, B. Varadarajan, Y. Shen, Y. Shen, Y. Chai, C. Schmid *et al.*, "TNT: Target-driven trajectory prediction," in *Conf. on Rob. Learning*. PMLR, 2021, pp. 895–904.

[19] E. A. Pool, J. F. Kooij, and D. M. Gavrila, "Crafted vs learned representations in predictive models—a case study on cyclist path prediction," *IEEE Trans. on Int. Vehicles*, vol. 6, no. 4, pp. 747–759, 2021.

[20] Y. Liu, Q. Yan, and A. Alahi, "Social NCE: Contrastive learning of socially-aware motion representations," in *Proc. of the IEEE/CVF Intl. Conf. on Comp. Vis.*, 2021, pp. 15 118–15 129.

[21] K. Mangalam, Y. An, H. Girase, and J. Malik, "From goals, waypoints & paths to long term human trajectory forecasting," in *Proc. of the IEEE/CVF Intl. Conf. on Comp. Vis.*, 2021, pp. 15 233–15 242.

[22] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social LSTM: Human trajectory prediction in crowded spaces," in *Proc. of the IEEE Conf. on CVPR*, 2016, pp. 961–971.

[23] X. Mo, Y. Xing, and C. Lv, "Heterogeneous edge-enhanced graph attention network for multi-agent trajectory prediction," *arXiv preprint arXiv:2106.07161*, 2021.

[24] J. F. Kooij, F. Flohr, E. A. Pool, and D. M. Gavrila, "Context-based path prediction for targets with switching dynamics," *Intl. J. of Comp. Vis.*, vol. 127, no. 3, pp. 239–262, 2019.

[25] N. Deo and M. M. Trivedi, "Trajectory forecasts in unknown environments conditioned on grid-based plans," *arXiv preprint arXiv:2001.00735*, 2020.

[26] W. Zeng, M. Liang, R. Liao, and R. Urtasun, "LaneRCNN: Distributed representations for graph-centric motion forecasting," in *2021 IEEE/RSJ Intl. Conf. on Int. Rob. and Systems (IROS)*. IEEE, 2021, pp. 532–539.

[27] T. Gilles, S. Sabatini, D. Tsishkou, B. Stanciulescu, and F. Moutarde, "THOMAS: Trajectory heatmap output with learned multi-agent sampling," in *Intl. Conf. on Learning Representations*, 2021.

[28] ——, "GOHOME: Graph-oriented heatmap output for future motion estimation," in *2022 Intl. Conf. on Rob. and Automation (ICRA)*. IEEE, 2022, pp. 9107–9114.

[29] R. Greer, N. Deo, and M. Trivedi, "Trajectory prediction in autonomous driving with a lane heading auxiliary loss," *IEEE Rob. and Automation Letters*, vol. 6, no. 3, pp. 4907–4914, 2021.

[30] J. Amirian, B. Zhang, F. V. Castro, J. J. Baldelomar, J.-B. Hayet, and J. Pettré, "OpenTraj: Assessing prediction complexity in human trajectories datasets," in *Proc. of the Asian Conf. on Comp. Vis.*, 2020.

[31] I. Vizzo, T. Guadagnino, B. Mersch, L. Wiesmann, J. Behley, and C. Stachniss, "Kiss-ICP: In defense of point-to-point ICP–simple, accurate, and robust registration if done the right way," *IEEE Rob. and Automation Letters*, vol. 8, no. 2, pp. 1029–1036, 2023.

[32] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.

[33] F. Lu and E. Milios, "Globally consistent range scan alignment for environment mapping," *Autonomous Rob.*, vol. 4, pp. 333–349, 1997.

[34] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, Y. Bengio *et al.*, "Graph attention networks," *stat*, vol. 1050, no. 20, pp. 10–48 550, 2017.

[35] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[36] B. D. Ziebart, A. L. Maas, J. A. Bagnell, A. K. Dey *et al.*, "Maximum entropy inverse reinforcement learning." in *Aaai*, vol. 8. Chicago, IL, USA, 2008, pp. 1433–1438.