Proceedings of the 2022 IEEE
International Conference on Robotics and Biomimetics
December 5-9, 2022, Xishuangbanna, China

Learning Robust Vehicle Navigation Policy Under Interference and Partial Sensor Failures

Zhe Liu, Kefan Jin, Yu Zhai and Yanzi Miao

***Abstract*— Autonomous driving has been a research hotspot for several years. However, most researchers focus on the ideal driving condition, where the sensor data is always accurate and clean. In reality, the urban environment is complex and the sensor data can easily be disturbed, which can lead to failures of existing driving approaches. In this paper, we present a multi-sensor fusion based navigation framework to achieve robust navigation under interference and partial sensor failures, which implicitly learns the reliability of each input information and introduces graph neural networks to reconstruct features for robust navigation policy learning. Experiments on the high-fidelity CARLA platform demonstrate the effectiveness of our approach, especially our strong robustness against various disturbances on both the input point cloud data and the visual images.**

I. INTRODUCTION

In the past ten years, autonomous driving related technologies have attracted a lot of attentions from both the industry and academic communities, and made great achievements on the aspects of perception, localization, mapping, planning and control [1], [2]. In recent years, researchers have paid increasing attentions to the learning-based navigation framework, which learns the vehicle control commands imitatively from the raw observation information and human/expert driving behaviors [3]. Learning-based navigation contributes to learn more efficient features and directly map them to the navigation policy.

Existing learning-based navigation approaches mostly focus on investigating the multi-sensor fusion issue [4], introducing more semantic and global information [5], increasing the safety and interpretability [6], or learning various relations and scene understanding [7]. However, they have lost sight of the external interference and sensor failure issues, which are un-neglected in complex environments such as industrial environment, mountain area, large-scale scene with bad weather conditions or adversarial scenarios. The robustness to such challenges is very critical and essential for the practical implementation of self-driving systems in real cases [8].

In order to resolve the robustness issue, traditional systems usually introduce additional safety modules to monitor the reliability of input information and system state, detect abnormal data and decrease the influence by calling the redundant standby systems [8]. However, such systems require huge efforts on parameter tuning, data processing, sub-module pre-training and testing, and are costly and inefficient.

Z. Liu and K. Jin are with the MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University, China. Y. Zhai and Y. Miao are with the with the School of Information and Control Engineering, China University of Mining and Technology, China. The first two authors contribute equally. Corresponding author: Y. Miao.

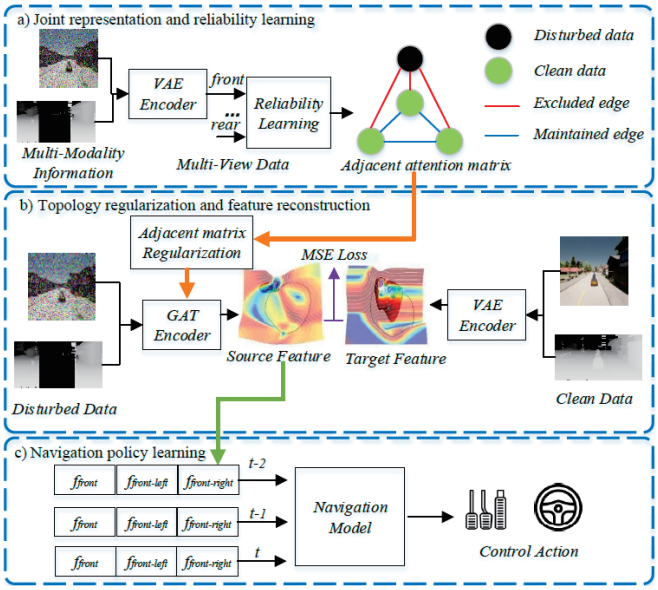

Fig. 1. The system framework of our robust navigation approach.

In order to solve the above issues, we investigate the robust navigation problem of self-driving vehicles in this paper, our contributions include:

- To the best of the authors' knowledge, we provide the first learning-based vehicle navigation framework which achieves robustness to external interference and partial sensor failures. We do not detect abnormal data explicitly, instead, we encourage the learning model to learn efficient representations of the input multi-sensor data, and to learn how to aggregate the information with uncertainties for robust navigation.
- The spatial-temporal multi-sensor information is aggregated by introducing the graph neural networks and an attention-based loss is designed to learn how to implicitly represent the reliability of each data. In addition, a topology regularization approach is formulated to re-build the information coupling structure and the abnormal data is reconstructed by using graph neural networks. Finally, a navigation module is developed to achieve the robust vehicle navigation task.
- We provide comprehensive evaluations on the professional vehicle simulator CARLA. Experimental results demonstrate our strong robustness to large sensor data noises and partial occlusions.

2022 IEEE International Conference on Robotics and Biomimetics (ROBIO) | 978-1-6654-8109-0/22/$31.00 ©2022 IEEE | DOI: 10.1109/ROBIO55434.2022.10011819

II. REVIEW

A. Learning-Based Vehicle Navigation

As mentioned before, learning-based navigation approaches have attracted increasing attentions in recent years. As a pioneer, Conditional Imitation Learning (CIL) [3] provides the first famous end-to-end vehicle navigation framework by only using the visual data. In order to fuse multi-modality sensor data, various fusion frameworks are designed [4]. Except the raw sensor data, high-level abstracted information and semantic information, such as the roadmap, traffic lights, speed limits, and bird-eye-view observations, are also considered in order to achieve reliable navigation performance in complex urban scenarios [5]. In addition, various relations are also considered, including the vehicle-vehicle relation [9], vehicle-road relation [10], vehicle-human relation [11], to comprehensively understand the whole driving scenario. For the navigation policy learning, both the imitation learning [12] and reinforcement learning [13] strategies are implemented. Furthermore, to improve the interpretability of the output commands and thus ensuring safety, direct control actions are replaced by the probabilistic control output [14] or trajectory over a horizon [6].

B. Abnormal Information Detection and Handling for Self-Driving Systems

Abnormal information detection, such as the trajectory uncertainty [15], fake sensor data [16], bad weather caused interference [17], partial sensor failure [18] and GPS attacks [19] are considered in the self-driving related researches. In order to handle the abnormal information, a commonly used strategy is to introduce monitoring and surveillance system [20], switch to the redundant standby systems [8] or ignore the abnormal data after failure detection [17]. In order to achieve the above tasks, a large number of redundant information is required [8], [18], which may be un-reasonable and un-accessible in practical applications. In addition, pre-training and tuning with specific dataset are required and huge efforts on pre-processing cannot be avoided [21].

C. Graph-Based Abnormal Information Detection and Reconstruction

Graph neural networks have been widely used to detect and reconstruct abnormal data in various application scenarios. To name a few, in [22], the authors replace raw data with event and build graph auto-encoder for anomaly detection and reconstruction. In [23], the authors introduce graph neural networks to interpolate un-sampled signal data given the observed signals from other related information during the same period. In [24], the authors determine attentions with the hypothesis that similar nodes are more likely to interact than dissimilar nodes, and then mitigate adverse effects by modifying the message passing operators of graph neural networks. In [25], the authors recover the clean graph structure from the noisy and perturbed graph by recovering the corresponding adjacency matrix to have the low rank and sparsity. More recently, in [26], the authors recover the disturbed visual observation by fusing the images from other view-points with graph neural networks.

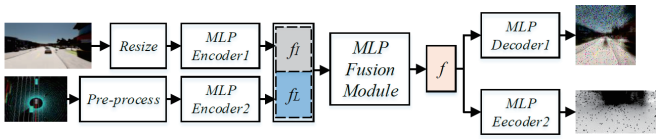

Fig. 2. Joint representation learning model.

III. MAIN APPROACH

A. System Framework

The overall system framework of our robust navigation approach is shown in Figure 1, which mainly includes three modules:

- *Joint Representation and Reliability Learning:* We first learn the joint representation of multi-sensor data by using the Variational Auto-Encoder (VAE) and then introduce the spatial-temporal graph to implicitly learn the reliability of each data, the output of this module is an adjacent attention matrix which penalizes the weights corresponding to the abnormal data related links.
- *Topology Regularization and Feature Reconstruction:* We then formulate the regularization approach to rebuild the information coupling structure, i.e., to further improve the adjacent attention matrix. The supervised-learning policy is utilized to reconstruct abnormal information to match the clean one.
- *Navigation Policy Learning:* Finally, we develop the learning module with the reconstructed data and output the vehicle control commands.

B. Joint Representation and Reliability Learning

We first implement the VAE structure to extract features from the input data. Most existing VAEs are single-input-sing-output, in this paper, we extend the model provided in [27] into the multi-input-multi-output version. As shown in Figure 2, the input image is resized and the input point cloud is transferred into a virtual image, and two MLPs are utilized to extract features f_I and f_L respectively. The concatenated feature $[f_I; f_L]$ is input into another MLP module to obtian the joint representation f. Two corresponding MLP-based decoders are implemented to reconstruct the visual image and virtual lidar image respectively. Except the original VAE loss, we also add the MSE losses of the two input channels in training.

Secondly, we introduce a data classification task to enable the model to recognize abnormal information by aggregating the spatial-temporal data. As shown in Figure 3, we build the spatial-temporal graph by connecting the data of all the viewpoint at the same time step and connecting the data of the same viewpoint at time-series steps. Graph Attention Layers (GALs) are utilized to aggregate information on the spatial-temporal graph and the final output is a score which indicates if the corresponding data is abnormal, i.e., the reliability of each input joint representation. Here we consider two losses,

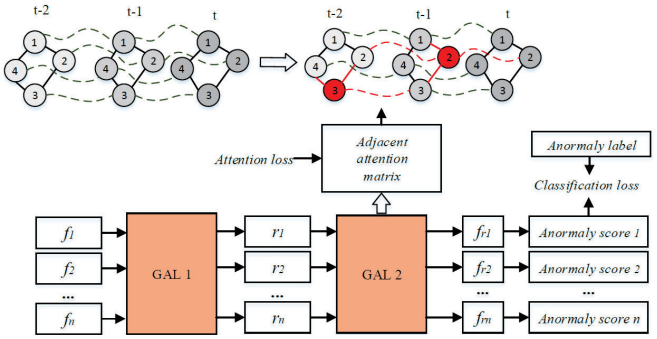

Fig. 3. Reliability learning model.

the first one is the classification loss and the second one is the attention loss, which is defined as:

$$attention\ loss: \sum_{e_{ij} \notin \mathcal{N}} a_{ij} - \sum_{e_{ij} \in \mathcal{N}} a_{ij}, \tag{1}$$

where $\mathcal{N}$ is the set of edge $\{e_{ij}\}$ that at least links one abnormal node on the spatial-temporal graph, a_{ij} is the weight (attention) of e_{ij}. The above attention loss enables the model to give higher weights to the normal information and lower weights to the abnormal information during graph aggregation. Please note that we only use the information classification as an auxiliary task, i.e., we will not use the output score or feature f_{ri} in the follow modules, instead, we will only use the adjacent attention matrix $\bar{A}$ (we assume the original adjacent matrix of the spatial-temporal graph is A).

C. Topology Regularization and Feature Reconstruction

In this section, we introduce the topology regularization approach to re-build the information coupling structure and re-construct the abnormal data simultaneously. More specifically, we formulate the topology regularization as follows:

$$\arg\min_{S} \mathcal{L}_S = ||S - \bar{A}||^2 + \alpha \cdot tr(F L_S F), \tag{2}$$

where S denotes the regularized adjacent matrix, A denotes the original obtained adjacent matrix and $\bar{A}$ is the output of the Joint Representation and Reliability Learning model. For the Laplacian regularization, F includes the input joint representations f of all the node in the spatial-temporal graph, L_S is the Laplacian matrix of S. Minimizing $\mathcal{L}_S$ will obtain a new adjacent matrix S which is re-built from $\bar{A}$ and is expected to link consistent information to filter out the abnormal data.

With the regularized Laplacian matrix S, we further build another graph which is similar with that in the Joint Representation and Reliability Learning model. The input is the same original input joint representations F but the output is the reconstructed features X (the source feature in Figure 1). In the meantime, we also use VAE encoder to generate the expected clean feature from the corresponding clean data (the target feature in Figure 1). Finally, MSE loss is utilized to learn the reconstruction features F.

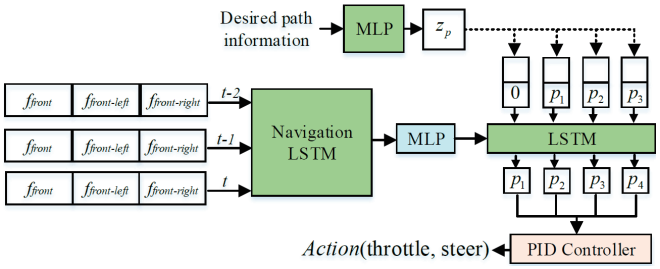

Fig. 4. Vehicle navigation model.

D. Navigation Policy Learning

In this section, we design the final navigation model to learn the vehicle control policy.

As shown in Figure 4, the input of this module is the reconstructed information X, in which here we only use the features f_{front} (front-view), $f_{front-left}$ (front-left-view), and $f_{front-right}$ (front-right-view). Three steps time-series data is considered. We first concatenate the data from three views and utilize a LSTM module and a MLP to aggregate the time-series data. The high-level desired path information (path waypoints) is input into another MLP to extract global path features z_p. Then we utilize another LSTM module, which input the aggregated features (as hidden features) as well as the global path features z_p, and output four future navigation points $p_1, ..., p_4$. Then a low-level PID controller is further implemented to generate the final control actions from the future navigation points.

We implement the imitation-learning strategy for navigation policy learning. The real driving trajectories of expert policy are collected in CARLA and then the ground-truth future navigation points can be generated from the expert trajectories, then the MSE loss is used to train the navigation model.

IV. IMPLEMENTATION

In this paper, the neural networks are constructed by using the Pytorch framework. The algorithms are implemented on a computer with an NVIDIA GeForce GTX 1070 GPU and an i5 10400 CPU.

In this paper, all the experiments are conducted in CARLA platform. Before training process, we collect 41000 training data samples in the environment Town01 provided in CARLA. Each data sample contains perceptual information in 8 views, including the front view, front-left view, left view, rear-left view, rear view, rear-right view, right view, and front-right view. Perceptual information in each view contains camera image data and lidar point cloud data. The sampling frequency in CARLA is set to 2 samples per simulation second. We use a training batch size of 24. All models were trained using Adam, and Xavier initialization is adopted to initialize the weights of the convolution layer. The evaluation experiments are conducted in the environment Town02 and Town05 provided in CARLA.

We employ the number of multiple perceptual layers l=2 with feature dimensions are 512 and 256 respectively in encoders for lidar point cloud data and camera image data.

The decoders of lidar data and image data have symmetrical structure. The reliability learning network consists of 2 graph attention layers with feature dimensions of 512 and 32 respectively. The feature reconstruction network consists of 2 graph attention layers with feature dimensions of 2048 and 1024 respectively. We employ the number of hidden size 512 in time-series data aggregating LSTM module and the number of hidden size 64 in time-series position predicting LSTM module. The learning rate is 0.00001.

The follow metrics are utilized to evaluate the driving performance of each model:

- *Route Completion (RC)*: The average number of the completed distance percentage R_i on all the N testing routes. Note that if the vehicle drives outside the route lanes, the off-road distance will not be included in the completed distance.

$$RC = \frac{1}{N}\sum_{i=1}^{N} R_i \tag{3}$$

- *Driving Score (DS)*: This index is the weighted average of the route completion with infraction multiplier P_i as described in [28] (P_i describes the overall driving performance on each route).

$$DS = \frac{1}{N}\sum_{i=1}^{N} R_i P_i \tag{4}$$

- *Infractions per km (IPK)*: The infractions include collisions with pedestrians, static facilities and other vehicles.

$$IPK = \frac{\sum_{i=1}^{N} infractions_i}{\sum_{i=1}^{N} k_i} \tag{5}$$

  where k_i denotes the completed distance (k_i km) in the i^{th} route.

In this paper, we consider the following baseline models for comparison and ablation study:

- *LateFusion*: This model is provided in [29] and we use this model as a baseline for comparison.
- *Our_Full* $\backslash w$ *RL*: In this model, we delete the reliability learning modules, i.e., we directly input the original adjacent matrix A into the adjacent matrix regularization module, which then generates the regularized adjacent matrix S for feature reconstruction to obtain X, and then the reconstructed features$X$ will be directly input to the final navigation policy learning module.
- *Our_Full* $\backslash w$ *LR*: In this model, we delete the topology regularization modules, i.e., we do not conduct the topology regularization presented in Equation (2). We directly use the output adjacent attention matrix $\bar{A}$ of Joint Representation and Reliability Learning module to initialize the GAT-based feature reconstruction module, and then the learned reconstructed features X will be directly input to the final navigation policy learning module.
- *Our_Full*: This is our full model described in the above section.

V. EXPERIMENTS

A. Robustness Evaluation

Two types of uncertainties are considered in this experiment, i.e., we manually block some parts both on the input visual image and virtual point cloud image to simulate the partial sensor failure case, and introduce random pepper&salt noises as well as image blurs to simulate the external interference. In testings, we increase the size of blocking area and the interference intensity with different levels. More specifically, we consider four disturbance levels: For the first level, in order to simulate external interference, we add 10 percents random pepper&salt noises in both the visual image and the virtual point cloud image (also introduce small image blurs), in order to simulate partial sensor failures, a random area of 10×20 pixel is blocked; For the fourth level, the random pepper&salt noises are increased to 70 percents (also introduce large image blurs) and a random area of 54×96 pixel is blocked. The second and third levels are between the above two extreme cases. In the testings, 50 percents data is disturbed by random pepper&salt noises and blurs (with different levels), while the rest 50 percents data is partially occluded.

We first evaluate the driving performance of *Our_Full* model under clean data. Comparing the results of *Our_Full* and *LateFusion* in the left part in Table I, we can see that our full model outperforms *LateFusion* on all the testing routes.

The right part of Table I demonstrates our robustness to various interference and sensor failures. Comparing the results of *Our_Full* and *LateFusion*, we can clearly see that our model maintains strong robustness in the presence of disturbances, however, the performance of *LateFusion* is destroyed.

Table II show the results under different disturbance levels, comparing *Our_Full* and *LateFusion* clearly shows that the performance of our model can be maintained during the increasing of the disturbance intensity. However, for *LateFusion* model, the driving score decreases rapidly as disturbance level increases.

B. Ablation Study

In this section, we provide ablation studies to demonstrate the effectiveness of each module in our robust navigation framework. All the baseline models are defined in Section IV and the results are shown in Table I and Table II. We can observe that: 1) Introducing topology regularization does improve the route completion rate RC. 2) Introducing the reliability learning will provide the navigation model a basic ability to address abnormal information. 3) The generalizability of *Our_Full* $\backslash w$ *RL* is relatively lower than our full model, as its performance in Town05(long) is much lower that in Town01(short). Note that compared to Town05, the scenarios in Town01 is more similar with out training environment Town02. 4) Our full model achieves the best performance.

TABLE I

DRIVING PERFORMANCE EVALUATIONS AND COMPARISONS

	clean data				disturbed data (level 4)					
	Town01(short)		Town05(long)		Town01(short)			Town05(long)		
	RC	DS	RC	DS	RC	DS	IPK	RC	DS	IPK
LateFusion	85.342	48.715	68.101	29.205	55.993	7.732	0.22	4.652	1.32	0.096
Our_Full \w RL	95.929	83.358	58.363	19.036	92.293	73.075	0.065	86.006	40.985	0.065
Our_Full \w LR	63.013	39.906	49.489	41.941	52.446	40.778	0.057	56.446	38.406	0.013
Our_Full	100	65.127	71.727	47.198	100	70.2	0.035	100	67.392	0.009

TABLE II

DRIVING PERFORMANCES UNDER VARIOUS DISTURBANCE LEVELS (UNDER TOWN05LONG)

	clean data		disturbance level 1		disturbance level 2		disturbance level 3		disturbance level 4	
	RC	DS	RC	DS	RC	DS	RC	DS	RC	DS
LateFusion	68.101	29.205	75.366	5.302	9.242	2.819	6.535	3.358	4.652	1.32
Our_Full \w RL	58.363	19.036	73.758	41.199	73.245	42.494	75.561	45.716	86.006	40.985
Our_Full \w LR	49.489	41.941	51.317	51.305	54.32	35.756	58.314	32.663	56.446	38.406
Our_Full	71.727	47.198	100	63.808	86.356	53.462	88.624	67.905	100	67.392

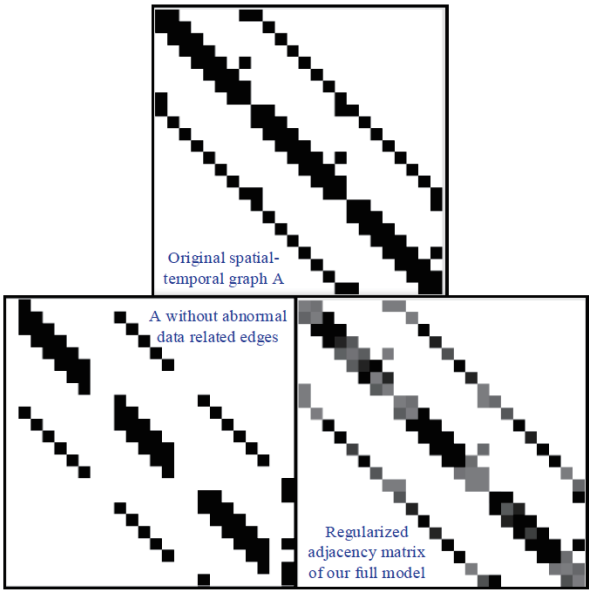

Fig. 5. Comparison results of the Laplacian regularization.

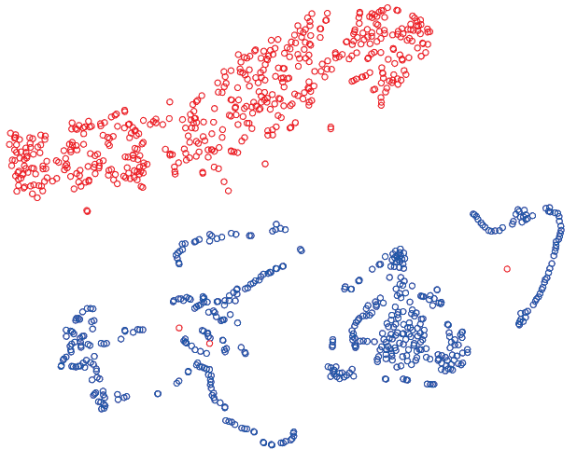

Fig. 6. The distribution of joint representations with and without disturbances (the red and blue points denote the joint representations with and without disturbances, respectively).

C. Visualization

In this section, we visualize of results for better understanding. We first visualize our Laplacian regularization results. As shown in Figure 5, we visualize the adjacency matrix A of the original spatial-temporal graph and the regularized adjacency matrix S in *Our_Full* model. In addition, we also visualize an updated version of the original adjacency matrix A by deleting all the edges which at least links one abnormal data (note that this is not a ground-truth as the abnormal data cannot be ignored completely). The results show that our regularized adjacency matrix S gives relative low attentions to the abnormal data successfully.

Then, we visualize the distribution of our joint representations of the clean data and the data with disturbance. Figure 6 shows that the two types of representations can be easily clustered in different regions.

Finally, we visualize our real driving scenarios in Figure 7, in which case we add disturbances in the visual images and add also add random pepper&salt noises in the front-right view point cloud data. *Our_Full* model can still maintain robust driving performance in this challenging scenario.

VI. CONCLUSIONS

In this paper, we provide a robust vehicle navigation approach. The experimental results validate our effectiveness to various external interference and partial sensor failures. Ablation studies demonstrate the effectiveness of each part

Fig. 7. Visualization of our real driving scenarios with interference and partial sensor failures.

in our framework and comparisons with existing works show our advanced robustness.

As all the sub-modules in our framework are differentiable, our approach can be further extended into a complete end-to-end version. In the future, we will consider more interference and even external attacks, and provide a fully end-to-end robustness navigation framework.

REFERENCES

[1] Z. Liu, C. Suo, Y. Liu, Y. Shen, Z. Qiao, H. Wei, S. Zhou, H. Li, X. Liang, H. Wang, and Y.-H. Liu, "Deep learning-based localization and perception systems: Approaches for autonomous cargo transportation vehicles in large-scale, semiclosed environments," *IEEE Robotics Automation Magazine*, vol. 27, no. 2, pp. 139–150, 2020.

[2] Z. Liu, S. Zhou, C. Suo, P. Yin, W. Chen, H. Wang, H. Li, and Y. Liu, "Lpd-net: 3d point cloud learning for large-scale place recognition and environment analysis," in *IEEE/CVF International Conference on Computer Vision*, 2019, pp. 2831–2840.

[3] F. Codevilla, M. Muller, A. Lopez, V. Koltun, and A. Dosovitskiy, "End-to-end driving via conditional imitation learning," in *IEEE International Conference on Robotics and Automation*, 2018, pp. 4693–4700.

[4] Y. Xiao and *et.al.*, "Multimodal end-to-end autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 1, pp. 537–547, 2022.

[5] M. Bansal, A. Krizhevsky, and A. Ogale, "Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst," in *Robotics: Science and Systems Conference*, 2019.

[6] W. Zeng and *et.al.*, "End-to-end interpretable neural motion planner," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8660–8669.

[7] S. Mylavarapu and *et.al.*, "Understanding dynamic scenes using graph convolution networks," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2020, pp. 8279–8286.

[8] R. Thakker and *et.al.*, "Autonomous off-road navigation over extreme terrains with perceptually-challenging conditions," in *International Symposium on Experimental Robotics*, 2021, pp. 161–173.

[9] X. Ma, J. Li, M. J. Kochenderfer, D. Isele, and K. Fujimura, "Reinforcement learning for autonomous driving with latent state inference and spatial-temporal relationships," in *IEEE International Conference on Robotics and Automation*, 2021, pp. 6064–6071.

[10] J. Pan and *et.al.*, "Lane-attention: Predicting vehicles moving trajectories by learning their attention over lanes," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2020, pp. 7949–7956.

[11] C. Chenand S. Huand P. Nikdeland G. Mori and M. Savva, "Relational graph learning for crowd navigation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2020, pp. 10 007–10 013.

[12] S. Hecker, D. Dai, A. Liniger, M. Hahner, and L. Van Gool, "Learning accurate and human-like driving using semantic maps and attention," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2020, pp. 2346–2353.

[13] H. Shi, L. Shi, M. Xu, and K. Hwang, "End-to-end navigation strategy with deep reinforcement learning for mobile robots," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 4, pp. 2393–2402, 2020.

[14] A. Amini, G. Rosman, S. Karaman, and D. Rus, "Variational end-to-end navigation and localization," in *International Conference on Robotics and Automation*, 2019, pp. 8958–8964.

[15] J. J. Q. Yu, "Sybil attack identification for crowdsourced navigation: A self-supervised deep learning approach," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4622–4634, 2021.

[16] Y. Caoand *et.al.*, "Adversarial sensor attack on lidar-based perception in autonomous driving," in *ACM SIGSAC Conference on Computer and Communications Security*, 2019, pp. 2267–2281.

[17] C. Zhang, Z. Huang, M. H. Ang, and D. Rus, "Lidar degradation quantification for autonomous driving in rain," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2021, pp. 3458–3464.

[18] G. Liu, A. Siravuru, S. Prabhakar, M. Veloso, and G. Kantor, "Learning end-to-end multimodal sensor policies for autonomous navigation," in *Conference on Robot Learning*, 2017, pp. 249–261.

[19] S. Narain, A. Ranganathan, and G. Noubir, "Security of gps/ins based on-road location tracking systems," in *IEEE Symposium on Security and Privacy*, 2019, pp. 587–601.

[20] Z. Liu, W. Chen, H. Wang, Y.-H. Liu, Y. Shen, and X. Fu, "A self-repairing algorithm with optimal repair path for maintaining motion synchronization of mobile robot network," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 3, pp. 815–828, 2020.

[21] J. Nitsch, M. Itkina, R. Senanayake, J. Nieto, M. Schmidt, R. Siegwart, M. J. Kochenderfer, and C. Cadena, "Out-of-distribution detection for automotive perception," in *IEEE International Intelligent Transportation Systems Conference*, 2021, pp. 2938–2943.

[22] H. Zhang, Z. Li, and Z. yu Ren, "Data-driven construction of data center graph of things for anomaly detection," *arXiv preprint arXiv:2205.15997*, 2020.

[23] Y. Wu, D. Zhuang, A. Labbe, and L. Sun, "Inductive graph neural networks for spatiotemporal kriging," in *AAAI Conference on Artificial Intelligence*, 2021, pp. 4478–4485.

[24] X. Zhang and M. Zitnik, "Gnnguard: Defending graph neural networks against adversarial attacks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9263–9275, 2020.

[25] W. Jin, Y. Ma, X. Liu, X. Tang, S. Wang, and J. Tang, "Graph structure learning for robust graph neural networks," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, p. 66–74.

[26] Y. Zhou, J. Xiao, Y. Zhou, and G. Loianno, "Multi-robot collaborative perception with graph neural networks," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2289–2296, 2022.

[27] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[28] K. Chitta, A. Prakash, B. Jaeger, Z. Yu, K. Renz, and A. Geiger, "Transfuser: Imitation with transformer-based sensor fusion for autonomous driving," *arXiv preprint arXiv:2205.15997*, 2022.

[29] I. Sobh and *et.al.*, "End-to-end multi-modal sensors fusion system for urban automated driving," in *Neural Information Processing Systems*, 2018.