Azam, Shoaib; Munir, Farzeen; Kyrki, Ville; Kucner, Tomasz Piotr; Jeon, Moongu; Pedrycz, Witold

Exploring Contextual Representation and Multi-modality for End-to-end Autonomous Driving

Published in:
Engineering Applications of Artificial Intelligence

DOI:
10.1016/j.engappai.2024.108767

Published: 01/09/2024

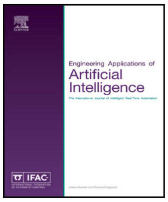
Document Version
Publisher's PDF, also known as Version of record

Please cite the original version:
Azam, S., Munir, F., Kyrki, V., Kucner, T. P., Jeon, M., & Pedrycz, W. (2024). Exploring Contextual Representation and Multi-modality for End-to-end Autonomous Driving. *Engineering Applications of Artificial Intelligence*, *135*, Article 108767. https://doi.org/10.1016/j.engappai.2024.108767

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai



Research paper

Exploring Contextual Representation and Multi-modality for End-to-end Autonomous Driving

Shoaib Azam [a,b,*], Farzeen Munir [a,b], Ville Kyrki [a,b], Tomasz Piotr Kucner [a,b], Moongu Jeon [c], Witold Pedrycz [d,e,f]

[a] Department of Electrical Engineering and Automation, Aalto University, Espoo, Finland
[b] Finnish Center for Artificial Intelligence, Finland
[c] School of Electrical Engineering and Computer Science, Gwangju Institute of Science and Technology, Gwangju, 61005, South Korea
[d] Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB T6R 2V4, Canada
[e] Department of Electrical and Computer Engineering, Faculty of Engineering, King Abdulaziz University, Jeddah, 21589, Saudi Arabia
[f] Systems Research Institute, Polish Academy of Sciences, Warsaw, 01-447, Poland

ARTICLE INFO

Keywords:
Vision-centric autonomous driving
Attention
Contextual representation
Imitation learning
Vision transformer

ABSTRACT

Learning contextual and spatial environmental representations enhances autonomous vehicle's hazard anticipation and decision-making in complex scenarios. Recent perception systems enhance spatial understanding with sensor fusion but often lack global environmental context. Humans, when driving, naturally employ neural maps that integrate various factors such as historical data, situational subtleties, and behavioral predictions of other road users to form a rich contextual understanding of their surroundings. This neural map-based comprehension is integral to making informed decisions on the road. In contrast, even with their significant advancements, autonomous systems have yet to fully harness this depth of human-like contextual understanding. Motivated by this, our work draws inspiration from human driving patterns and seeks to formalize the sensor fusion approach within an end-to-end autonomous driving framework. We introduce a framework that integrates three cameras (left, right, and center) to emulate the human field of view, coupled with top-down bird-eye-view semantic data to enhance contextual representation. The sensor data is fused and encoded using a self-attention mechanism, leading to an auto-regressive waypoint prediction module. We treat feature representation as a sequential problem, employing a vision transformer to distill the contextual interplay between sensor modalities. The efficacy of the proposed method is experimentally evaluated in both open and closed-loop settings. Our method achieves displacement error by 0.67 m in open-loop settings, surpassing current methods by 6.9% on the nuScenes dataset. In closed-loop evaluations on CARLA's Town05 Long and Longest6 benchmarks, the proposed method enhances driving performance, route completion, and reduces infractions.

1. Introduction

The autonomous driving ecosystem involves the perception and planning modules to complement each other for a smooth course of action (Yurtsever et al., 2020). These systems, fundamental to the autonomous driving ecosystem, are tasked with interpreting vast amounts of sensory data to understand the vehicle's surroundings and make real-time decisions to navigate complex environments safely. To this end, two approaches—modular (Azam et al., 2020) and end-to-end autonomous driving (Xiao et al., 2020; Khan et al., 2022)—have been adopted in academia and industry as possible solutions for perception and planning modules. While the modular approach offers the advantage of interpretability and modular debugging, it is often criticized for its potential bottlenecks in data processing and decision latency. On the other hand, end-to-end autonomous driving presents a promising alternative, offering scalability and the potential to directly learn driving policies from raw sensory inputs, thereby providing a more streamlined integration of perception and action (Schwarting et al., 2018).

Addressing the dynamic and unpredictable nature of driving environments is paramount for advancing end-to-end autonomous driving systems. While existing strategies that use various sensor modalities, such as single-camera systems and LiDAR, have significantly contributed to capturing environmental details, they often fall short in dynamically adapting to the rapidly changing context of real-world scenarios. Several techniques have been developed to extract spatial

[*] Corresponding author at: Department of Electrical Engineering and Automation, Aalto University, Espoo, Finland.
E-mail address: shoaib.azam@aalto.fi (S. Azam).

https://doi.org/10.1016/j.engappai.2024.108767
Received 2 January 2024; Received in revised form 16 April 2024; Accepted 31 May 2024
Available online 10 June 2024

and temporal information from these modalities (Huang et al., 2020; Behl et al., 2020). Among these, sensor fusion techniques have marked a leap forward in creating a more holistic understanding of the vehicle's surroundings. Yet, the challenge persists in achieving an adaptable and subtle perception that can prioritize the shifting relevance of environmental features. The interactions between multiple dynamic agents and the need for a comprehensive representation that spans different views or modalities highlight the critical gap in current methodologies: the ability to maintain a global contextual awareness amidst the complexity and unpredictability of the driving environment. As mentioned earlier, the limitations underscore the necessity for an adaptive approach that synthesizes spatial and temporal information from multi-modal sensory inputs and intelligently adapts to the evolving context of the environment, mirroring the adaptability and situational awareness akin to human perception and decision-making processes.

To address the above-mentioned limitations, we introduced a novel end-to-end encoder–decoder framework for predicting waypoints as illustrated in Fig. 1. Our work aims to simulate a human-like approach to perception and decision-making by incorporating immediate visual data with a global understanding of the environment. This dual-modality approach sets our framework apart from existing methodologies by improving decision-making capabilities under various driving conditions. To this end, our approach is the integration of multi-camera views (left, right, and center) with top-down Bird's-Eye View (BEV) semantic maps through a transformer-based encoder. A key feature of our framework is incorporating a self-attention mechanism within the encoder. This mechanism empowers the system to intelligently adjust its prioritization of environmental features based on their immediate relevance and contextual significance. This capability is essential for navigating scenarios where the importance of environmental elements can shift rapidly, providing a solution to the previously static interpretation of sensor data and fulfilling the need for a deeper, better understanding of the environment.

The proposed framework is structured around two key components: (i) the perception module and (ii) the waypoint prediction module. The perception module extracts features from multi-camera views and BEV (Bird's-Eye View) semantic maps through a dedicated backbone network. Subsequently, these features undergo a fusion process, after which they are fed into a transformer network. This network is responsible for refining the feature representation, ensuring a comprehensive and cohesive understanding of the vehicle's surroundings. Following this feature enhancement, the information is relayed to a Gated Recurrent Unit (GRU)-based waypoint prediction module tasked with generating the navigational waypoints. The rationale behind proposing this unique framework and encoder stems from our objective to simulate a more human-like approach to perception and decision-making in autonomous driving. Just as a human driver integrates immediate visual cues with an overarching understanding of their environment, our model synthesizes data from RGB cameras with BEV semantic maps to form a comprehensive and adaptable representation of the surroundings. By doing so, our approach aims to surmount the limitations of existing methods, offering superior navigation and decision-making capabilities that are robust across various driving conditions.

To validate the effectiveness of our proposed method, we have conducted extensive experimental analysis in both open-loop and closed-loop settings. In open-loop settings, we have evaluated the performance of our method in terms of Euclidean distance (L2 norm) using the nuScenes dataset, and it has surpassed the state-of-the-art methods. Moreover, we have employed two Carla benchmarks, Town05 Long and Longest6, to assess the performance of our method in closed-loop settings. Our method has demonstrated superior performance in terms of driving, route completion, and infraction score compared to the state-of-the-art methods, reinforcing its robustness across various driving conditions.

In summary, our work has following contributions:

1. Designing a framework that demonstrates an integration of spatial perception through RGB cameras with a top-down bird's-eye view (BEV) for contextual mapping. This dual approach mimics human-like perception by combining immediate visual data with a global understanding of the environment, enhancing the autonomous system's ability to navigate complex scenarios
2. Develop a transformer-based encoder to sequence the spatial and contextual features, leading to an improved feature representation for learning the driving policies.

The remainder of this paper is structured as follows: Section 2 provides a review of the relevant literature. The problem formulation is discussed in Section 3. The proposed framework is detailed in Section 4, while Section 5 is dedicated to the experimental setup, analysis, and results. Section 6 focuses on the ablation studies conducted. Section 7 delves into discussions and outlines directions for future research, and Section 8 offers concluding remarks for the paper.

2. Related work

2.1. Multi-modal end-to-end learning frameworks for autonomous driving

Learning optimal trajectories involve a better representation of the environment to include spatial, temporal, and contextual information of the environment. Different multi-modal end-to-end driving methods are developed in the literature to improve driving performance. These multi-modal methods either use cameras, Lidar, HD maps, or sensor fusion between these information modalities. Xiao et al. (2020) have used the sensor fusion between RGB cameras and depth information to investigate the use of multi-modal data compared to single modality for end-to-end autonomous driving. Some works have focused on semantics and depth for determining the explicit intermediate representation of the environment and their effect on autonomous driving (Behl et al., 2020; Zhou et al., 2019). In addition, some works, for instance, NMP (Zeng et al., 2019), have used the Lidar and HD maps first to generate the intermediate 3D detections of the actors in the future and then learn a cost volume for choosing the best trajectory. Lidar and camera fusion are extensively used for perception and obtaining driving policies. Sobh et al. (2018) have used the Lidar and image fusion by processing both sensor modality streams in a separate branch and then fusing the resulting features. Further, they have applied semantic segmentation and Lidar post-processing Post Grid Mapping to increase the method's robustness. Similarly, Prakash et al. (2021) have fused the Lidar and camera data at multiple levels through self-attention for learning the driving policies. In addition, some methods have adopted sensor fusion between camera and semantic maps (Natan and Miura, 2022) for learning end-to-end driving policy for autonomous driving. Several studies have investigated the application of knowledge distillation techniques to learn driving policies. In this approach, a privileged agent is initially trained with access to comprehensive information, such as maps, navigational data, and images. Subsequently, this privileged agent is employed to train a sensorimotor agent, which only has access to image data (Chen et al., 2020b; Zhang et al., 2023). Furthermore, improving the decoder architecture in an encoder–decoder architecture is also being explored by Jia et al. (2023). All these methods have used sensor fusion techniques to acquire the spatial or temporal information of the environment but lack contextual information in terms of BEV semantic maps. In the proposed work, we have opted for BEV semantic maps and incorporated them with a camera stream to answer whether the inclusion of BEV semantic maps improves driving performance.

2.2. BEV representation end-to-end autonomous driving

Representing the environment in a BEV benefits the planning and control task as it circumvents the issues like occlusion and scale distortion and also provides the contextual representation of the environment. In this context, some works focus on generating the BEV representation; for instance, ST-P3 leverages spatial–temporal learning by designing an egocentric-aligned representation of BEV and finally uses that representation for perception, planning, and control (Hu et al., 2022). Hu et al. (2023) have designed an end-to-end planning autonomous driving framework. This framework's perception and prediction modules are structured as transformer decoders, with task queries acting as the interface between these two nodes. An attention-based planner is used to sample the future waypoints by considering the past node's data. Following the same approach, Jiang et al. (2023) have used a vectorized representation for end-to-end autonomous driving. They have adopted a BEV encoder for BEV feature extraction combined with map and agent queries in a transformer network for environment representation and then a planning transformer for predicting the trajectories. In addition, Chitta et al. (2021) have proposed a neural attention field for waypoint prediction. All these methods have used the BEV representation, similar to our work, but are more focused on how to make the BEV representation from input images; however, in our work, we focus on how to use the BEV features for learning the policy rather make BEV from input images and then use it for the planning. The experimental analysis shows the efficacy of our proposed method against state-of-the-art methods illustrating the effectiveness of using BEV representation for learning the driving policies in both open and closed-loop settings.

2.3. Sensor fusion for autonomous driving

In autonomous driving, the integration of various sensor types—such as cameras, LiDAR, and radar—is crucial for a comprehensive understanding of the vehicle's environment, as each sensor type has distinct limitations that can be mitigated through combined usage (Yurtsever et al., 2020; Huang et al., 2022). In this aspect, multi-modal sensor fusion has become the preferred approach (Chen et al., 2020a, 2017; Fadadu et al., 2022; Meyer et al., 2020). In literature, sensor fusion is typically classified according to the stage at which multi-modal data fusion occurs during the feature representation learning process. Notably, three primary fusion strategies, early, late, and intermediate-level fusion approaches, are studied in the research (Tang et al., 2023; Munir et al., 2023).

Recent studies in multi-modal end-to-end autonomous driving perform sensor fusion between RGB cameras, LiDAR, depth and semantic data, and radar to enhance driving performance. For instance, (Haris and Glowacz, 2022; Codevilla et al., 2018; Huang et al., 2020) have employed an early fusion approach to fuse multi-modal data to learn the driving policies. Similarly, in the case of late fusion approaches, multi-modal data is fused at the decision level for learning the driving policies using multi-modalities as proposed in this works (Huang et al., 2023). However, neuroscience research indicates that intermediate-level fusion can enhance feature representations learned from multiple modalities, offering a more comprehensive understanding of the environment (Schroeder and Foxe, 2005; Macaluso, 2006). In this work, we have adopted this approach to fuse the intermediate features from multi-view cameras and BEV semantic maps to learn the driving policies. In literature, most approaches have followed this approach; for instance, LAV designed a framework that learns from the behaviors of all observed vehicles, not just the ego-vehicle (Chen and Krähenbühl, 2022). LAV fuses the RGB and LiDAR data to represent the environment using PointPainting (Vora et al., 2020), combining the semantic information extracted from the RGB with the LiDAR point cloud. Similarly, Confuse fuses the RGB and LiDAR feature maps to learn better feature representation at different levels (Liang et al., 2018). In addition to fusing data between LiDAR and cameras, another promising approach is to generate BEV maps/features. In this regard, extracting features from multi-modal input and converting them into a shared BEV space can then be utilized for downstream tasks (Man et al., 2023; Liu et al., 2023).

Similarly, some methods have utilized transformer-based approaches for fusing the multi-modal data at intermediate levels for learning driving policies (Singh, 2023; Ye et al., 2023). Initially used for natural language processing tasks (Vaswani et al., 2017), transformers have widely been employed for learning meaningful representation in vision applications (Dosovitskiy et al., 2020; Carion et al., 2020). The transformer's self-attention module enhances the learning of sequential data globally and improves feature representation. Prakash et al. (2021) employed the transformer to combine intermediate features representation from RGB images and Lidar data. Huang et al. (2022) design a transformer-based neural prediction framework that considers social interactions between different agents and generates possible trajectories for autonomous vehicles. Dong et al. (2021) determines the driving direction from visual features acquired from images by using a novel framework consisting of a visual transformer. The driving directions are decoded for human interpretability to provide insight into learned features of the framework. Finally, (Li et al., 2020) considers social interaction between agents on the road and forecasts their future motion. The spatial–temporal dependencies were captured using a recurrent neural network combined with a transformer encoder. The closest to our work is (Shao et al., 2022), which uses transformer-based encoder–decoder architecture with safety constraints. However, we believe that using the vision transformer learns the structure of the fused features independently, attending to the most relevant parts of the features to make predictions, and can produce high-quality intermediate representations. In contrast, traditional transformer-based encoder–decoders are less efficient in capturing the global dependencies in the features.

3. Problem formulation

In this work, an end-to-end learning approach is adopted for the point-to-point navigation problem, where the objective of the trained agent is to safely reach the goal point by learning a driving policy π^* that imitates the expert policy π. The learned policy completes the given route by avoiding obstacles and complying with the traffic rules. In the closed-loop settings, we have opted for the CARLA simulator to collect the expert dataset in a supervised learning approach. Similarly, to use the expert data in open-loop settings, we have used the nuScenes dataset. Suppose the dataset $D = (X^j, Y^j)_{j=1}^{d}$ of size d is collected that consists of high dimensional observations vector X from the sensory modalities along with the corresponding expert trajectories vector Y. The expert trajectories are defined in vehicle local coordinate space and are set of 2D waypoints transformed that is, $Y = \mathbf{y_t} = (u_t, v_t)_{t=1}^{T}$, where u_t and v_t are the position information in horizontal and vertical directions, and T corresponds to the future horizon for the waypoints, respectively. The objective is to learn the policy π with the collected dataset D in a supervised learning framework with the loss function $\mathcal{L}$ expressed as follows

$$\arg\min_{\pi} \mathbb{E}_{(X,Y)\to D}[\mathcal{L}(Y, \pi(X))]. \tag{1}$$

In this urban setting, the high-dimensional observations include the center, right, left cameras and top-down BEV semantic data.

4. Method

Fig. 1 illustrates the overview of the proposed method, which comprises two main components: (i) the perception module and (ii) the waypoint prediction modules. The perception module extracts features from input sensor modalities and then forwards them to the waypoint prediction module to generate future waypoints/trajectories. The following sections details the perception and waypoint prediction modules.

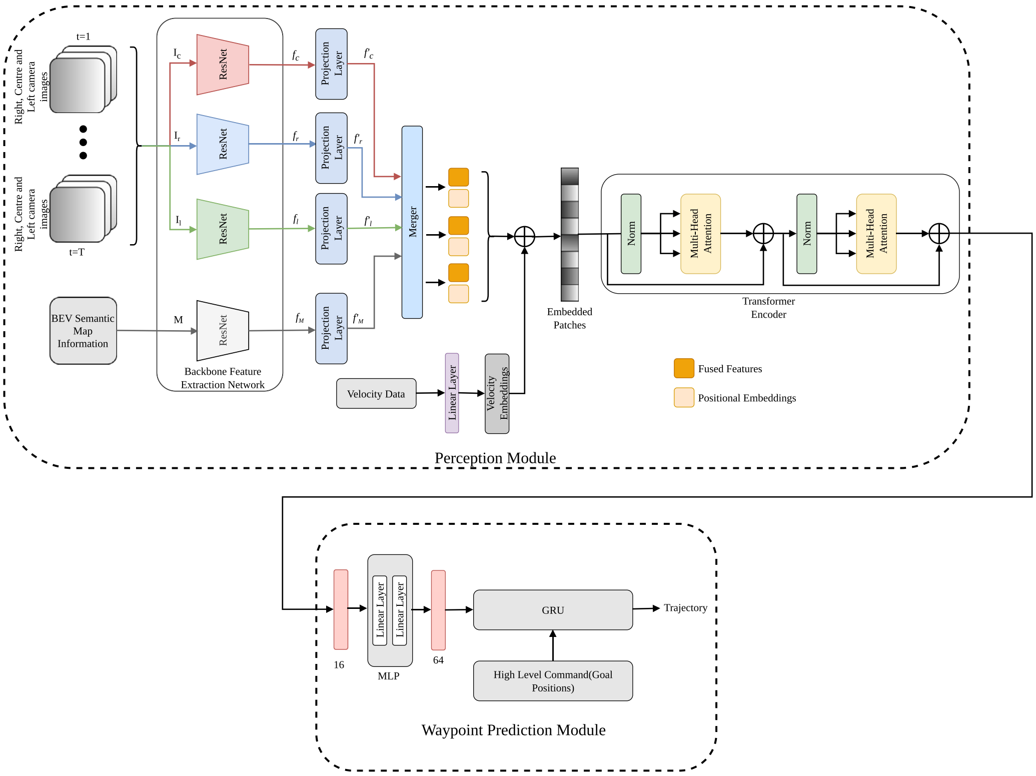


Fig. 1. Architecture of a Proposed Multi-modal Perception and Waypoint Prediction framework for End-to-End Autonomous Driving: The proposed method's architecture is composed of two main components: the Perception Module and the Waypoint Prediction Module. In the Perception Module, features are extracted from three RGB camera inputs and BEV semantic maps, which are then enhanced with velocity data before being processed by a transformer encoder. The encoded features are subsequently relayed to the Waypoint Prediction Module, where a GRU (Gated Recurrent Unit) predicts the vehicle's future waypoints based on these features. (Best view in color).

4.1. Perception module

The perception module incorporates a backbone network designed to extract features from sensor modalities, coupled with a transformer-based encoder network for representation learning.

(A) Backbone Feature Extraction Network: In this module of the proposed method, we aim to construct a spatio-temporal representation of the environment. The sequence of input RGB images from three distinct views that are center($I_c \in \mathcal{R}^{3\times H\times W}$), right ($I_r \in \mathcal{R}^{3\times H\times W}$), and left ($I_l \in \mathcal{R}^{3\times H\times W}$)—with width W and height H, are processed through a feature extraction backbone network. For the feature extraction network, we have adopted ResNet(50) architecture; however, in our ablation studies, we have evaluated the other variants of ResNet architecture to validate the performance of the proposed method. Similarly, BEV (Bird's-Eye View) semantic maps ($M \in \mathcal{R}^{H\times W}$), which encode spatial layout and environmental context, are integrated into the perception pipeline to complement the image-derived features. Suppose I_v, where v corresponds to the three views (I_c, I_l, and I_r) and BEV semantic maps (M), is passed to the backbone network to extract the features maps f_v for each sensor modalities expressed as:

$$f_v = \mathcal{F}(I_v; \theta_{\mathcal{F}}) \tag{2}$$

where $\theta_{\mathcal{F}}$ encapsulates the trainable parameters of the feature extraction network. After extracting the feature maps f_v, from the backbone feature extraction network, it is necessary to synthesize these into a single representation that encapsulates spatio-temporal information of the environment. To this end, a project layer $\mathcal{P}$ is employed that converts the feature maps f_v of all sensor modalities to low dimensional feature maps f'_v for all the sensor modalities. Mathematically, as expressed in Eq. (3),

$$f'_v = \mathcal{P}(f_v; \theta_{\mathcal{P}}) \tag{3}$$

where, in our experiment, we have kept the size of this low dimension feature maps to 400 for all feature maps f_v, respectively. $\theta_{\mathcal{P}}$ denotes the weights associated with the projection layer, which are learned during the training process to optimize the fusion of features. To encapsulate the features representation to a unified embedding, all the feature maps f'_v from three views and BEV semantic maps are merged in conjunction with vehicle velocity data through linear transformation layers $\mathcal{L}$ as shown in Eq. (4)

$$f = \mathcal{L}(Concat(f'_c, f'_l, f'_r, f'_M); \theta_{\mathcal{L}}) \tag{4}$$

Finally, to make combined feature maps f compatible as input to the transformer encoder, we have used post-processing techniques to reshape the feature maps f from the 1600 dimension to $(B, 1, 40, 40)$ dimension.

(B) Transformer Encoder: In this work, a transformer encoder, specifically a vision transformer, is employed to learn the contextual relationship between the features and to generalize it to learn better feature representation. In this context, the resulting features $f = \mathcal{R}^{1\times H\times W}$ is fed to the transformer encoder by flattening into patches $f_p = \mathcal{R}^{N\times(P^2C)}$, where H and W corresponds to the resolution of input features from the backbone network, C is the number of channels, (P, P) is the size of each patch, and $N = HW/P$ denotes the number of patches and also the input sequence length. In addition, a learnable position embedding is added to the input sequence, a trainable parameter with the same dimension as the input sequence, so that the network infers the spatial dependencies between different tokens at the train time. A velocity embedding is also added to the C dimensional of the input sequence through a linear layer, which includes the current velocity. Finally, the input sequence, positional embeddings E_{pos}, and velocity embeddings E_{vel} are element-wise summed together, which is mathematically

expressed in the following,

$$\begin{aligned}
&z_o = [f_p^1 E; f_p^2 E; \cdots; f_p^N E] + E_{pos} + E_{vel},\\
&E \in \mathcal{R}^{(P^2.C)\times D},\\
&E_{pos} \in \mathcal{R}^{(N+1)\times D}, E_{vel} \in \mathcal{R}^{(N+1)\times D},\\
&z_l' = MSA(LN(z_{l-1})) + z_{l-1} + z_{l-1},\\
&z_l = MLP(LN(z_l')) + z_l' + z_l',
\end{aligned} \tag{5}$$

where MSA corresponds to multi-head self-attention, MLP is multi-layer perceptron, LN is layer normalization, and D corresponds to dimension. The multi-head attention helps in generating the rich feature representation for the input sensor modalities that in turn to learn better contextual representation. The formulation of the multi-head self-attention is expressed as,

$$\begin{aligned}
&(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \mathbf{z}\mathbf{W}_{QKV},\\
&\mathbf{W}_{QKV} \in \mathcal{R}^{D\times 3D_h},\\
&A = softmax(\mathbf{Q}\mathbf{K}^{\mathbf{T}})/\sqrt{D_h},\\
&A \in \mathcal{R}^{N\times N},\\
&SA(\mathbf{z}) = A\mathbf{v},\\
&MSA(\mathbf{z}) = [SA_1(\mathbf{z}); SA_2(\mathbf{z}); \cdots; SA_j(\mathbf{z})]\mathbf{W}_{msa},\\
&\mathbf{W}_{msa} \in \mathcal{R}^{(j.D_h)\times D}.
\end{aligned} \tag{6}$$

where $\mathbf{Q}$, $\mathbf{V}$ and $\mathbf{K}$ are the query, value and key vectors and $\mathbf{W}$ is the weight matrix. The output features from the MSA have the same dimensionality as the input features. The transformer encoder applies the attention multiple times throughout the architecture. The final output features from the transformer encoder are then summed along the dimension to produce the 16 dimensional vector having the contextual representation of features from all the sensor modalities. This resulting 16 dimensional feature vector is injected into the waypoint prediction module to predict waypoints.

4.2. Waypoint prediction module

The waypoint prediction module acts as a decoder for predicting future waypoints using the encoded information from the transformer encoder. The resulting 16 dimensional vector is passed through an MLP consisting of two hidden layers having 256 and 128 units, respectively, to output the 64 dimensional vector. The MLP layer is used for upsampling the vector dimension from 16 to 64 and is related to experimental heuristics that produce better results in terms of waypoint prediction. We have employed the auto-regressive GRU model to predict the next waypoints that take the 64 dimension feature vector to initiate the hidden state of the GRU model. The GRU-based auto-regressive model takes the current position and goal location as high-level commands as input, which helps the network focus on the relevant context in the hidden states to predict the next waypoints. In the case of closed-loop settings, the goal locations include the GPS points registered in the same ego-vehicle coordinate frame as input to the GRU rather than the encoder because of the colinear BEV space between the predicted waypoints and the goal locations. However, high-level commands such as forward, turn right and left are passed as input to the GRU for waypoint predictions in the open-loop settings.

In the open-loop settings, we have evaluated the predicted trajectory with the ground-truth trajectory without using a controller. However, for the closed-loop setting, the predicted waypoints are passed to the control module of the CARLA simulator to generate steer, throttle, and brake values. Two PID controllers for lateral and longitudinal control are used in this context. The longitudinal controller takes the average weighted magnitude of vectors between the waypoints of consecutive time steps, whereas the lateral control takes their orientation.

5. Experiments

This section explains the proposed method evaluation in both open-loop and closed-loop settings. The nuScenes dataset is utilized for the open-loop evaluation, whereas the CARLA simulator is used for the closed-loop evaluation.

5.1. Open-loop experiments on nuscenes

(A) Dataset: The nuScenes dataset contains 1k diverse scenes comprising different weather and traffic conditions. Each scene is 20 seconds long and contains 40 frames, corresponding to a total of 40k samples in the dataset. The dataset is recorded using a camera rig comprised of 6 cameras on ego-vehicle, giving a full 360 deg view of the environment. The dataset includes the calibrated intrinsic K and extrinsic (R, t) for each camera view at every time-step. The proposed method settings utilize the center, right, and left camera views. Since the nuScenes dataset does not provide any top-down BEV semantic representations, the BEV semantic representation is generated using ego-vehicle poses and camera views intrinsic and extrinsic calibration data.

(B) Input Representations: For the nuScenes dataset, the input image from the center, front, and left camera views are first cropped and resized to 256 × 256 from the original resolutions of 900 × 1600. Contrary to the camera views and ego-vehicle future positions data, the nuScenes data does not provide the top-down BEV semantic maps. Given the necessity of BEV semantic maps for the proposed method, we follow the off-shelf Cross-view Transformer method (Zhou and Krähenbühl, 2022) to generate the BEV semantic maps. It is to be noted, that one can use better alternatives to generate the richer BEV semantic maps. The Cross-view Transformer utilizes the encoder–decoder architecture to achieve precise map-view semantic segmentation. An image encoder generates the multi-scale feature maps for each image view. Later, these feature maps are merged using cross-view attention into cohesive map-view representation. The cross-view attention relies on positional embeddings attuned to the scene's geometric layout, facilitating accurate alignment between camera and map views. All camera views shared the same image encoder, each employing camera-specific positional embeddings on their individual camera calibration parameters. Finally, a lightweight convolutional decoder upsamples the refined map-view embedding and produces the final segmentation output. In our settings, we kept the resolution of this BEV semantic map to 256 × 256.

(C) Output Representations: The proposed method predicts the future trajectory Y, for the ego-vehicle in the ego-vehicle coordinate. In the open-loop settings, the future trajectory Y is represented as waypoints that include position information. In our experiments, by default, the horizon $T = 2.0$ s is set for predicting the future trajectory by taking the past 1.0 s past context.

(D) Evaluation Metrics: For the proposed method evaluation, Euclidean distance (L2 error) is used which is the measure of distance between the expert trajectory and the predicted trajectory. Mathematically, the L2 error is defined by the Eq. (7)

$$L2(T_e, T_p) = \sum_{i=1}^{n} \sum_{j=1}^{d} (T_{eij} - T_{pij})^2, \tag{7}$$

where, T_e and T_p correspond to the expert and predicted trajectory, respectively. Each trajectory consists of n points in a d-dimensional space.

5.2. Closed-loop experiments on CARLA

(A) Dataset: In this work, CARLA 0.9.10[1] simulator is used to create a dataset for training and evaluation. Table 1 illustrates the dataset

[1] https://carla.org/

Table 1
Dataset generation details using the CARLA simulator for the proposed method.

Maps	Training Data: Town01, Town02, Town03, Town04, Town06, Town07, Town10 Test Data: Town05
Weather Conditions	Clear sunset, Clear noon, Wet noon, Wet sunset, Cloudy noon, Cloudy sunset, Rainy noon, Rainy sunset
Non-player characters (NPCs)	Pedestrians, Car, Bicycle, Truck, Motorbike
Object Classes	0:Unlabeled, 1:Pedestrian, 2:Road line, 3:Road, 4:Sidewalk, 5:Car, 6:Red traffic light, 7: Yellow traffic light, 8: Green traffic light
Routes	Tiny (only straight or one turn) Short (100–500 m) Long (1000–2000 m)
CARLA Version	0.9.10

details that are utilized in generating the training dataset to create a more varying simulation environment. For generating the dataset, an expert policy with the privileged information from the simulation is rolled out to save the data at 2FPS. The dataset includes left, right, and center camera RGB images, top-down semantic map information, the corresponding expert trajectory, speed data, and vehicular controls. The trajectory includes 2D waypoints transformed into BEV space in the vehicle's local coordinate, whereas the steering, throttle, and brake data are incorporated into the vehicular control data at the time of recording. Inspired by Prakash et al. (2021) configurations, we have gathered the data by giving a set of predefined routes to the expert in driving the ego-vehicle. The GPS coordinates define the routes provided by the global planner and high-level navigational commands (e.g., turn right, follow the lane, etc.). We have generated around 60 hours of the dataset, including $200K$ frames.

(B) Input Representation: The proposed method utilizes two modalities: RGB cameras (left, center and right) and semantic maps. The three RGB cameras provide a complete field of view that mimics the human field of view. The semantic maps are converted to BEV representation that contains ground-truth lane information, location, and status of traffic lights, vehicles, and pedestrians in the vicinity of ego-vehicle. The top-down semantic maps are cropped to the resolution of 256 × 256 pixels. For all three cameras, to cater the radial distortion, the resolution is cropped to 256 × 256 from the original camera's resolution of 400 × 300 pixels at the time of extracting the data.

(C) Output Representation: For the point-to-point navigation task, the proposed method predicts the future trajectory Y of the ego-vehicle in the vehicle coordinate space. The future trajectory Y is represented by a sequence of $2D$ waypoints, $Y = \mathbf{y_t} = (u_t, v_t)_{t=1}^{T}$, where u_t and v_t are the position information in horizontal and vertical directions, respectively. In the experimental analysis, we have utilized $T = 4$ as the number of waypoints.

(D) Evaluation Metrics: The proposed method's efficacy is evaluated using the following metrics indicated by the CARLA driving benchmarks.

Route Completion: is the percentage of route distance R_j completed by the agent in route j averaged across the number of N routes is shown in the form,

$$RC = \frac{1}{N}\sum_{j}^{N} R_j. \tag{8}$$

The RC is reduced if the agent drives off the specified route by some percentage of the route. This reduction in RC is defined by a multiplier (1-% off route distance).

Infraction Multiplier: as shown in (9) is defined as the geometric series of infraction penalty coefficient, p^i, for every infraction encountered by the agent along the route. Initially, the agent starts with the ideal base score of 1.0, which is reduced by a penalty coefficient for every infraction. The penalty coefficient p^i for each infraction is predefined. If the agent collides with the pedestrian $p_{pedestrian}$, the penalty is set to 0.50; with other vehicles $p_{vehicles}$, it is set to 0.60, 0.65 for collision with static layout p_{stat}, and 0.7 if the agent breaks the red light p_{red}. The penalty coefficient is defined as $PC = p_{pedestrian}, p_{vehicles}, p_{stat}, p_{red}$,

$$IM = \prod_{i}^{PC}(p^i)^{infractions^i}. \tag{9}$$

Driving Score: is computed by taking the product between the percentage of the route completed by the agent R_j and the infraction multiplier IM_j of the route j and averaged by the number of the routes N_r. Higher driving score corresponds to the better model. Mathematically, the driving score (DS) is

$$DS = \frac{1}{N_r}\sum_{j=1}^{N_r} RC_j IM_j (p^i)^{infractions^i}. \tag{10}$$

It is to be noted that if the ego vehicle deviates from the route j for more than 30 meters or there is no action for 180 seconds, then the evaluation process on route j will be stopped to save the computations cost and next route will be selected for the evaluation process.

5.3. Training details

The proposed method is trained using the dataset collected from the CARLA simulator by rolling out the expert model and also on the nuScenes dataset. In addition, we have used the pre-trained ResNet50 model trained on the ImageNet dataset to extract the features in the backbone network for each sensor modality. In training the proposed network, we have added augmentation such as rotating and noise injection to the training data, along with adjusting the waypoints labels. For the transformer encoder, we have used the patch size of 4, which gives the 16 dimensional feature embedding. In addition we have adopted the attention layer of 12 in the transformer encoder. We have trained the proposed method using the Pytorch library on RTX 3090 having 24 GB GPU memory for a total of 100 epochs. In training, we have used the batch size of 64 and an initial learning rate of 10^{-4}, which is reduced by a factor of 10 after every 20 epochs. The L_1 loss function is used for training the proposed method. Let y_t^{gt} represent the ground-truth waypoints from the expert for the timestep t; then the loss function is represented as

$$\mathcal{L} = \sum_{t=1}^{T}\left\| y_t - y_t^{gt} \right\|_1. \tag{11}$$

An AdamW optimizer is used in training with a weight decay set to 0.01 and beta values to the Pytorch defaults of 0.9 and 0.99 (Yao et al., 2021).

5.4. Results

(A) Open-loop Experimental Results on nuScenes: The proposed method is evaluated on the L2 evaluation metric against the state-of-the-art methods for the quantitative analysis, as illustrated in Table 2. In our

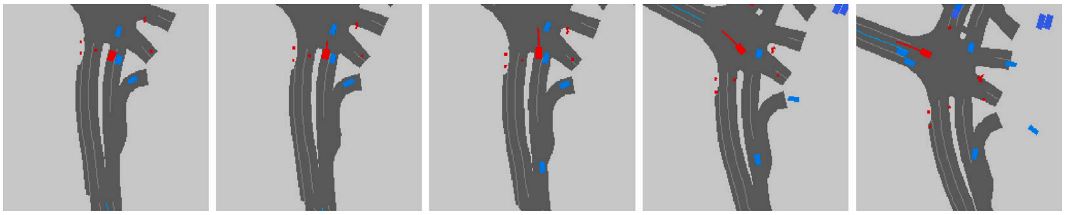


Fig. 2. Qualitative results for the proposed method in different driving conditions using nuScenes dataset in open-loop evaluation.

Table 2
Quantitative Comparison of Proposed Method in Open-loop Settings: The comparative analysis lists the L2 error between the proposed method and state-of-the-art methods for forecasted positions at 1 s, 2 s, and 3 s horizons using nuScenes dataset. A lower L2 error indicates better performance.

Method	L2 (m)			
	1s	2s	3s	Avg
NMP (Zeng et al., 2019)	–	–	3.18	–
FF (Hu et al., 2021)	0.55	1.20	2.54	1.43
ST-P3 (Hu et al., 2022)	1.33	2.11	2.90	2.11
UniAD (Hu et al., 2023)	0.48	0.96	1.65	1.03
VAD-Base (Jiang et al., 2023)	0.41	0.70	1.05	0.72
Ours	**0.35**	**0.61**	**1.01**	**0.66**

experiments, we have deactivated the ego-status information in the open-loop settings and also fixed the planning horizon $T = 3.0$ s to make a fair comparison with the state-of-the-art methods. Since the L2 error corresponds to the displacement error in meters between the predicted and ground-truth trajectories, the lower the displacement error, the better the model. The proposed method illustrates better performance as compared to state-of-the-art methods. The comparative analysis uses camera-centric and Lidar-based end-to-end learning methods to predict trajectory. For instance, NMP uses the Lidar and HD maps for predicting future trajectories, giving the L2 error of 3.18 m. Then NMP model is only evaluated for the planning horizon of 3.0 s. Similarly, the FF method predicts the future trajectory based on free-space estimation having the L2 error of 2.54 m at the planning horizon of 3.0 s and an average L2 error of 1.43 m. The proposed method illustrates lower L2 error at the planning horizon of 3.0 s and on average compared to NMP and FF methods. Similar to our work, the baseline methods that follow the BEV representation are ST-P3, UniAD, and VAD-Base. The L2 error for the ST-P3, UniAD, and VAD-Base are 2.90 m, 1.65 m, and 1.05 m, respectively, at the planning horizon of 3.0 s, where the proposed method has L2 error of 1.01 m at the same planning horizon, outperforming the ST-P3, UniAD, and VAD-Base by 89.5%, 38.8%, and 3.8% respectively. Similarly, on average, the proposed method shows lower L2 error than the state-of-the-art methods.

Fig. 2 illustrates the qualitative analysis of the proposed method when evaluated on the nuScenes dataset.

(B) Closed-loop Experimental Results on CARLA: We compare the proposed method with other state-of-the-art methods on two CARLA benchmarks, Town05 Long and Longest6, in closed-loop settings. Our quantitative analysis considers the baselines with multi-modality inputs rather than sticking with methods involving only a single modality. Using contextual information, the proposed method achieves better driving, route completion, and infraction scores. Table 3 illustrates the quantitative results of the proposed method on the Town05 Long benchmark. Specifically, the proposed method achieves the driving score of 68.30±1.90, 96.5±1.18 of route completion, and 0.75±0.05 of infraction score, outperforming the ThinkTwice by 4.8% in driving score, 1.03% in route completion and 8% in infraction score respectively. Similarly, the proposed method illustrates better evaluation metrics scores when compared with other state-of-the-art methods.

Table 4 shows the proposed method results with other state-of-the-art methods on the Longest6 benchmark in closed-loop settings. The proposed method achieves the driving score of 67.43 ± 2.3, 80.54 ± 1.5 of route completion, and 0.81 ± 0.05 of infraction scores on the Longest6 benchmark, outperforming the other state-of-the-art methods in evaluation metrics in the closed-loop settings. Figs. 3 and 4 illustrate the proposed method's qualitative results on Town05 and Longest6 benchmarks in various driving scenarios. The learned driving policy through the proposed method is displayed in moving straight, stopping at the traffic light, and making left, and right turns. These results demonstrate that the driving policy learned using the proposed method show promising results and complements the quantitative analysis of the proposed method with other state-of-the-art baseline methods.

6. Ablation studies

In this section, we further investigate the performance of the proposed method by conducting ablation studies that explore the impact of different BEV map generation techniques on our approach, as well as examining the influence of various components on overall performance.

6.1. Comparative analysis of BEV semantic map generation techniques on proposed method

In our proposed method, we have utilized an off-the-shelf Cross-view Transformer (CVT) (Zhou and Krähenbühl, 2022) to create BEV semantic maps under open-loop settings. This study extends our investigation into how varying BEV map generation methods impact the performance of our approach in these settings, providing insights into the adaptability and effectiveness of different strategies.

To this end, we have employed Lift, Splat (Philion and Fidler, 2020), ST-P3 (Hu et al., 2022), BEVFormer (Li et al., 2022), and CVT (Zhou and Krähenbühl, 2022),BEV semantic map generation methods and used for the proposed method in predicting the waypoints in open-loop settings. Table 5 illustrates the quantitative results for the proposed method when used with different BEV generation methods. In our findings, the proposed method produces better results with CVT than other state-of-the-art methods. The Lift, Splat method shows increasing error over time. The ST-P3 has lower errors than Lift, Splat, while BEVFormer and CVT show significant improvements. CVT has the lowest errors across all time frames, indicating the highest accuracy when used with the proposed method for waypoint prediction.

Within the closed-loop context of our study, we have leveraged BEV semantic maps generated by the Carla simulator as part of our proposed methodology. However, alternative methods are viable for generating these maps. To examine the influence of BEV map quality on the performance of the proposed method within closed-loop settings, we have incorporated the ST-P3 (Hu et al., 2022) method to produce BEV semantic maps for the Town05 Long Benchmark. These maps were then input into our model. Table 6 presents a quantitative comparison, showcasing how different BEV map generation approaches affect our proposed method's performance in closed-loop scenarios.

Table 3

Quantitative Comparison of Proposed method in Closed-loop Settings for Town05 Long Benchmark: The analysis illustrates the quantitative comparison of proposed method with state-of-the-art methods on Town05 Long benchmark. The evaluation uses three metrics:driving score (DS), route completion (RC) and infraction score (IS).

Methods	Metrics		
	DS ↑	RC ↑	IS ↑
CILRS (Codevilla et al., 2019; Jia et al., 2023)	7.80 ± 0.30	10.30 ± 0.00	0.75 ± 0.05
LBC (Chen et al., 2020b; Jia et al., 2023)	12.30 ± 2.00	31.90 ± 2.20	0.66 ± 0.02
Transfuser (Prakash et al., 2021; Jia et al., 2023)	31.00 ± 3.60	47.50 ± 5.30	0.77 ± 0.04
SDC[a] (Natan and Miura, 2022)	47.13 ± 5.27	77.42 ± 0.00	0.65 ± 0.00
SDC[b] (Natan and Miura, 2022)	31.05 ± 2.70	64.13 ± 0.00	0.53 ± 0.00
Roach (Zhang et al., 2021; Jia et al., 2023)	41.60 ± 1.80	96.40 ± 2.10	0.43 ± 0.03
LAV (Chen and Krähenbühl, 2022; Jia et al., 2023)	46.50 ± 2.30	69.80 ± 2.30	0.73 ± 0.02
InterFuser (Shao et al., 2023)	51.60 ± 3.40	88.90 ± 2.50	0.58 ± 0.05
TCP (Wu et al., 2022; Jia et al., 2023)	57.20 ± 1.50	80.40 ± 1.50	0.73 ± 0.02
Think Twice (Jia et al., 2023)	65.00 ± 1.70	95.50 ± 2.00	0.69 ± 0.05
Ours	**68.30 ± 1.90**	**96.50 ± 1.18**	**0.75 ± 0.05**

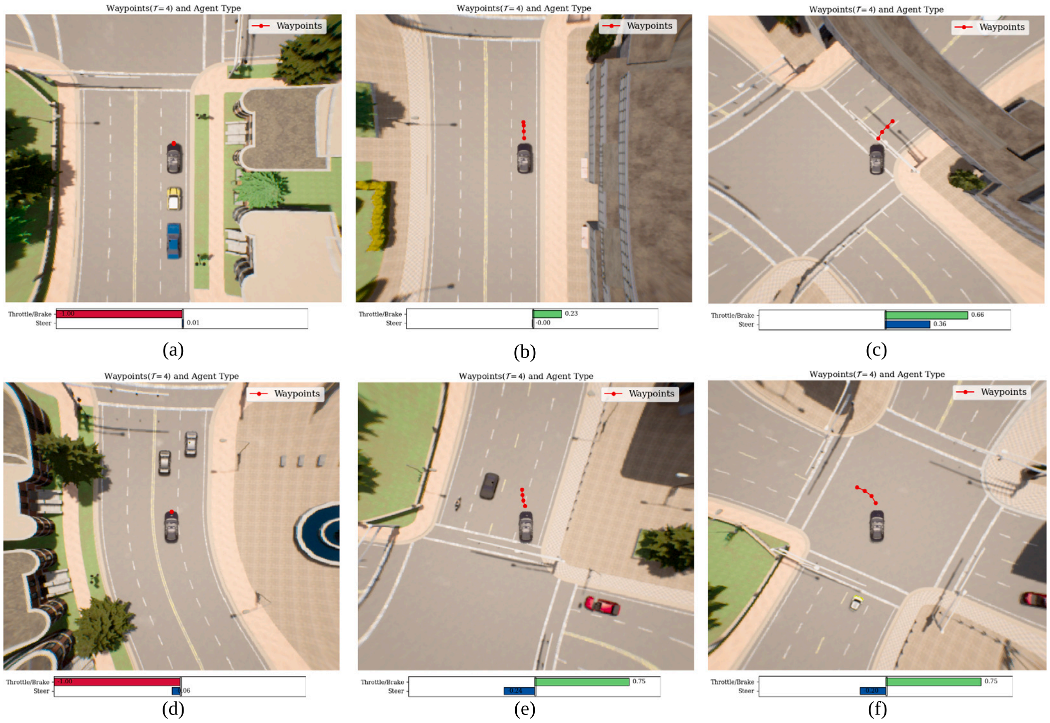
[a] Indicates the respective method reports the score on normal all weather conditions.

[b] Corresponds to adversarial all weather conditions.

Table 4

Comparative analysis of Proposed method in Closed-loop Settings for Longest6 Benchmark: The quantitative results show the comparison of proposed method with state-of-the-art methods on Longest6 benchmark in terms of driving score (DS), route completion (RC) and infraction score (IS).

Methods	Metrics		
	DS ↑	RC ↑	IS ↑
WOR (Chen et al., 2021; Zhang et al., 2023)	17.36 ± 2.95	43.46 ± 2.99	0.54 ± 0.06
LAV (Chen and Krähenbühl, 2022; Zhang et al., 2023)	48.41 ± 3.40	80.71 ± 0.84	0.60 ± 0.04
Transfuser (Prakash et al., 2021; Zhang et al., 2023)	46.20 ± 2.57	83.61 ± 1.16	0.57 ± 0.00
NEAT (Chitta et al., 2021; Zhang et al., 2023)	24.08 ± 3.30	59.94 ± 0.50	0.49 ± 0.02
CAT (Zhang et al., 2023)	58.36 ± 2.24	78.79 ± 1.50	0.77 ± 0.02
TCP (Wu et al., 2022; Zhang et al., 2023)	42.86 ± 0.63	61.83 ± 4.19	0.71 ± 0.04
Think Twice (Jia et al., 2023)	66.7	77.2	0.84
Ours	**67.43 ± 2.3**	**80.54 ± 1.5**	**0.81 ± 0.05**



(a) (b) (c) (d) (e) (f)

Fig. 3. Visualization of Proposed Method's Decision-making on Town05 Long Benchmark: The qualitative results illustrate the proposed method efficacy in different driving conditions (a–f) of Town05 Long benchmark. The results also highlights the throttle/brake and steer values of proposed method's action in different driving conditions.

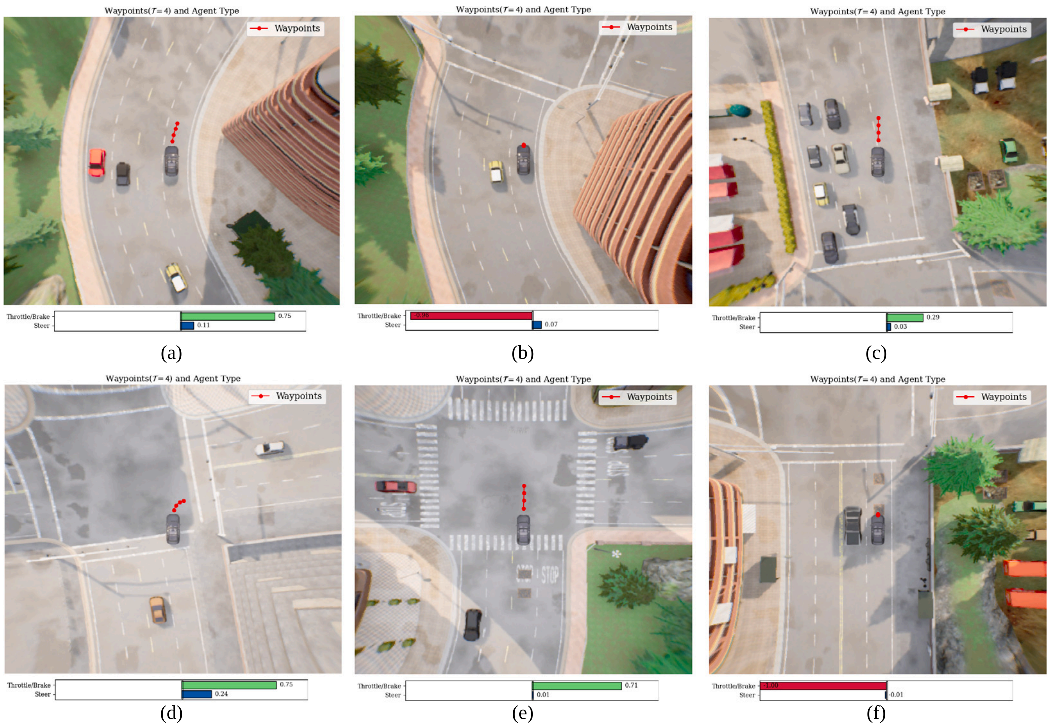

Waypoints($\mathcal{T}$= 4) and Agent Type
Waypoints
Throttle/Brake
Steer
0.75
0.11
(a)
Waypoints($\mathcal{T}$= 4) and Agent Type
Waypoints
Throttle/Brake
Steer
-0.96
0.07
(b)
Waypoints($\mathcal{T}$= 4) and Agent Type
Waypoints
Throttle/Brake
Steer
0.29
0.03
(c)
Waypoints($\mathcal{T}$= 4) and Agent Type
Waypoints
Throttle/Brake
Steer
0.75
0.24
(d)
Waypoints($\mathcal{T}$= 4) and Agent Type
Waypoints
Throttle/Brake
Steer
0.71
0.01
(e)
Waypoints($\mathcal{T}$= 4) and Agent Type
Waypoints
Throttle/Brake
Steer
-1.00
-0.01
(f)

Fig. 4. Visualization of Proposed Method's Decision-making on Longest6 Benchmark: The qualitative results showcase the proposed method's efficacy in different driving conditions with also throttle/brake and steer action values (a–f).

Table 5
BEV Map Generation: Quantitative Open-Loop Performance Analysis: The quantitative comparison illustrates the BEV map generation methods on proposed method's performance in an open-loop settings. A lower L2 error indicates better performance. All the maps generated from BEV methods are used with proposed method.

Method	L2 (m) ↓			
	1s	2s	3s	Avg
Lift, Splat Philion and Fidler (2020)	1.75	2.15	2.95	2.28
ST-P3 (Hu et al., 2022)	1.25	1.95	2.54	1.91
BEVFormer (Li et al., 2022)	0.40	0.85	1.49	0.91
CVT (Zhou and Krähenbühl, 2022)	**0.35**	**0.61**	**1.01**	**0.66**

Table 6
Closed-Loop Evaluation: BEV Map Impact on Proposed Method Performance: The comparative analysis shows the impact of BEV map generation methods on proposed method's performance in an closed-loop settings. All the maps generated from BEV methods are used with proposed method.

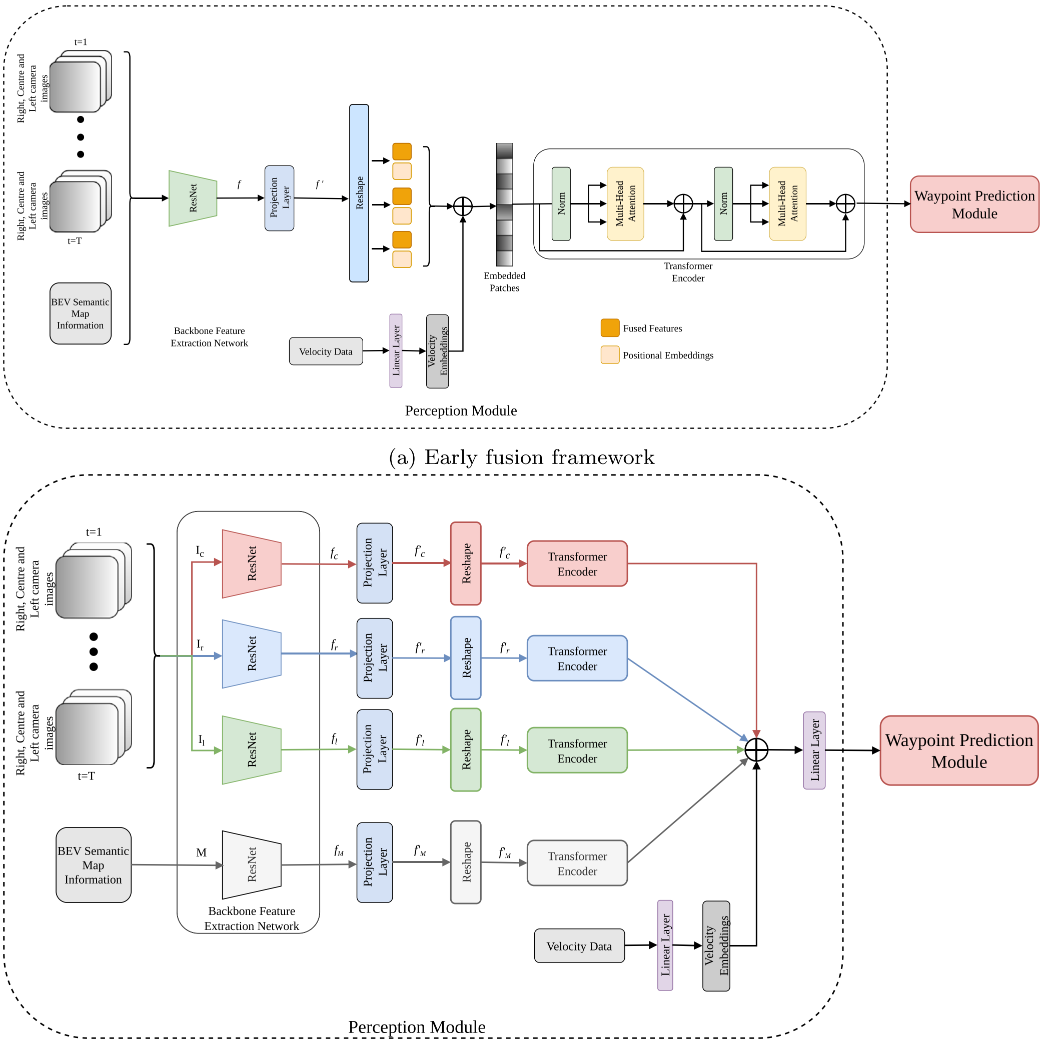
Method	Metrics		
	DS ↑	RC ↑	IS ↑
ST-P3 (Hu et al., 2022)	59.25 ± 2.40	87.54 ± 1.25	0.65 ± 0.03
Ours	**68.30** ± **1.90**	**96.5** ± **1.18**	**0.75** ± **0.05**

6.2. Comparative early and late fusion approaches with proposed method

Sensor fusion techniques utilize three principal paradigms to integrate multi-modal data: early, late, and intermediate fusion approaches. As our model employs the intermediate fusion paradigm, we have designed the early and late fusion approaches and conducted a comparative analysis against our proposed method. In the early fusion approach, as illustrated in Fig. 5(a), the multi-view camera and BEV semantic maps are stacked together before performing any feature extraction. At the network level, we have used a single ResNet module as a backbone network to extract the features, which are then fed to the Transformer encoder through the projection layer. In the late fusion approach, as shown in Fig. 5(b), we have adopted a uni-modal architecture for each sensor modality for the feature representation. For each multi-view RGB image and BEV semantic map, the ResNet network is used as a backbone network for the feature representation. Each feature is subjected to average pooling and flattening, reducing the dimensionality to a 512-dimensional vector. A projection layer transforms these vectors into 400-dimensional vectors suitable for input into the transformer encoder. A dedicated transformer encoder individually processes each transformed feature vector. This step emphasizes the learning of contextual relationships within each sensor modality. Following encoding, a late fusion technique is employed to concatenate these feature vectors with velocity embedding followed by a linear layer to make the 16 dimensional vector to be used by the waypoint prediction module. It is to be noted that in both early and late fusion, the transformer encoder and waypoint prediction module follow the same architecture as designed in the proposed method.

In our comparative analysis, we have evaluated both early and late fusion variants of the proposed method in open and closed-loop settings. Within the open-loop configuration as illustrated in Table 7, the early fusion variant exhibited an L2 error of 1.55 m, 2.25 m, and 2.99 m for the 1 s, 2 s, and 3 s prediction horizons, respectively. However, The proposed method, with its superior performance, demonstrated greater efficacy in the open-loop setting than the early fusion variant. Similarly, the late fusion approach has an L2 error of 0.42 m at the 1 s horizon, 0.74 m at the 2 s horizon, and 1.15 m at the 3 s horizon, which is much better than the early fusion approach but slightly comparable with the proposed method.

For the closed-loop settings, we have followed the same experimental protocols of the proposed method for early and late fusion approaches. In this regard, we have experimentally evaluated the early and late fusion approaches on Town05 Long and Longest6 benchmarks for closed-loop settings. On the Town05 Long benchmark, the early fusion approach has attained the driving score of 52.74±3.85, 80.12±2.50 for route completion, and an infraction score of 0.65±0.03, respectively. Similarly, the late fusion approach has obtained 60.35±1.50, 87.45±2.00, and 0.72 ± 0.05 of driving, route completion, and infraction scores, respectively. The proposed method has illustrated better driving, route completion, and infraction scores than early and late fusion approaches



(a) Early fusion framework

(b) Late fusion framework

Fig. 5. Comparative frameworks: (a) Early fusion approach (b) Late fusion approach.

Table 7
Fusion Approach Efficacy in Open-Loop nuScenes Analysis: The comparative analysis illustrates the quantitative results of proposed method's fusion approach and its early and late fusion variants on nuScenes in open-loop settings.

Methods	L2 (m)			
	1 s	2 s	3 s	Avg
Ours(Early Fusion)	1.55	2.25	2.99	2.26
Ours(Late Fusion)	0.42	0.74	1.15	0.77
Ours	**0.35**	**0.61**	**1.01**	**0.66**

Table 8
Fusion Variant Performance Comparison on Town05 Long Benchmark: The quantitative results indicate the proposed method's fusion approach with its early and late fusion variants on Town05 Long benchmark in closed-loop settings. The comparative analysis demonstrates the efficacy of the proposed method over its early and late fusion counterparts.

Methods	Metrics		
	DS ↑	RC ↑	IS ↑
Ours(Early Fusion)	52.74 ± 3.85	80.12 ± 2.50	0.65 ± 0.03
Ours(Late Fusion)	60.35 ± 1.50	87.45 ± 2.00	0.72 ± 0.05
Ours	**68.30 ± 1.90**	**96.50 ± 1.18**	**0.75 ± 0.05**

on the Town05 Long benchmark, as illustrated in Table 8. Additionally, Table 9 provides a quantitative comparison of the early and late fusion approaches against our proposed method on the Longest6 benchmark. The early fusion approach has achieved the driving score of 47.35±1.65, 65.50 ± 3.20 of route completion, and 0.69 ± 0.05 of infraction score, respectively. The late fusion approach on the Longest6 benchmark has obtained the 59.15 ± 2.00, 71.73 ± 3.50, and 0.77 ± 0.02 of driving, route completion, and infraction scores, respectively. The comparative analysis showcases that the proposed method illustrates better performance in terms of driving, route completion, and infraction score on the Longest6 benchmark against both the fusion approaches.

6.3. Effect of backbone architectures and attention layers on waypoint prediction

This section analyzes how different network components influence the final waypoint prediction accuracy in the proposed method and its early and late fusion variants. For the proposed method, the ResNet50 architecture is utilized to extract features from multi-view RGB cameras and Bird's Eye View (BEV) semantic maps. To assess the impact of varying backbone architectures, we implemented ResNet-34 and ResNet-18 models for feature extraction. The performance outcomes of employing ResNet-18 and ResNet-34 as the backbone in open-loop configurations

Table 9
Fusion Variant Performance Comparison on Longest6 Benchmark: The proposed method's fusion approach is quantitative compared with its early and late fusion variants on Longest6 benchmark for closed-loop settings. The quantitative analysis demonstrates the efficacy of the proposed method over its early and late fusion counterparts.

Methods	Metrics		
	DS ↑	RC ↑	IS ↑
Ours(Early Fusion)	47.35 ± 1.65	65.50 ± 3.20	0.69 ± 0.05
Ours(Late Fusion)	59.15 ± 2.00	71.73 ± 3.50	0.77 ± 0.02
Ours	**67.43 ± 2.3**	**80.54 ± 1.5**	**0.81 ± 0.05**

are detailed in Table 10. Additionally, the results from closed-loop scenarios on the Town05-Long and Longest6 benchmarks, with ResNet-18 and ResNet-34 backbones, are documented in Tables 11 and 12, respectively. Our comparative analysis demonstrates that alterations in the backbone network architecture markedly affect the efficacy of the proposed method. Moreover, we have also performed the analysis of different components of architecture for early and late fusion approaches. Table 10 illustrates the open-loop settings results with different backbones for the proposed method's early and late fusion variants. Similarly, the performance in closed-loop settings for different backbone networks on the Town05 Long and Longest6 benchmarks applied to both early and late fusion approaches is presented in Tables 11 and 12, respectively.

In addition to different backbone effects on the proposed and its early and late fusion variants, we have also analyzed how different attention layers affect the performance of waypoint prediction. For this purpose, in our study we have selected 2, 6, 8 and 12 attention layers for the transformer encoder and analyze the effect of those attention layers on the waypoint prediction's performance. For the proposed method, the configuration with 12 attention layers was established as yielding the most favorable results in both open-loop and closed-loop scenarios, as indicated in Table 10, Table 11, and Table 12. Similarly, we have also performed the comparative analysis for the early and late fusion approaches as illustrated in Table 11, and Table 12 respectively for both open-loop and closed-loop settings. The analysis led to the conclusion that the attention layers play a pivotal role in the learning of feature dependencies, which is crucial for accurate waypoint prediction.

7. Discussion about real-world application and future work

In this section, we delve into the integration of our proposed method within real-world applications and explore potential avenues for future research building upon our framework. Our proposed framework, developed and validated within a simulated environment, demonstrates significant potential for real-world application in autonomous driving systems. The simulation-based approach offers a controlled setting to rigorously test the system's capabilities and robustness under diverse conditions that can be challenging to replicate in the real world. To bridge the gap between simulation and real-world deployment strategies such as data augmentation and domain adaptation is pivotal, enabling the model to reflect the intricacies of real-world scenarios better. A notable challenge in leveraging our model, which relies on multi-camera views and BEV (Bird's-Eye View) semantic maps, is the real-time acquisition of BEV maps. The reliance on off-the-shelf BEV generation methods poses a significant computational hurdle. A viable solution to circumvent this issue involves the utilization of vector maps to provide an efficient BEV representation of the surroundings. Additionally, the integration of a safety feedback layer is vital for the precise translation of waypoint predictions into actionable commands for vehicle actuators, ensuring actions are executed safely and effectively. This safety layer, serving as an indispensable link between high-level decision-making and actuator-level execution, significantly enhances the framework's utility and dependability for real-world autonomous driving applications.

Future research endeavors present exciting prospects for enhancing the framework's perception capabilities and driving policy predictions. While the current model leverages RGB cameras and BEV semantic maps for environmental perception, incorporating additional sensing modalities such as radar and LiDAR could significantly enrich the perception module. This expansion would bolster the system's environmental awareness and its ability to navigate complex driving scenarios with increased accuracy and safety. Another promising area of exploration involves refining the contextual representation of the environment, mainly through the integration with neural network-based controllers that provide an extra layer of safety for deploying the currently proposed method to real-world implementation. These potential research directions could further enhance the framework's capabilities and contribute to the advancement of autonomous driving systems.

8. Conclusion

In this work, we explore the use of contextual information for learning driving policies in an end-to-end manner for autonomous driving. Drawing inspiration from the human neural map representation of the environment, we employ three RGB cameras coupled with a top-down semantic map to achieve a holistic understanding of the surroundings. This environmental representation is then channeled through a self-attention-based perception module, subsequently processed by a GRU-based waypoint prediction module for generating the waypoints. The proposed method is experimentally evaluated for both open-loop and closed-loop settings, illustrating better performance than state-of-the-art methods. Moreover, to underscore the proficiency of our proposed technique, we have conducted ablation studies to evaluate how various elements of the architecture influence waypoint forecasting accuracy and the impact of different BEV generation methods on the proposed method's efficacy. Likewise, we have examined the ramifications of implementing early and late fusion strategies as variants of the proposed method on waypoint prediction outcomes.

CRediT authorship contribution statement

Shoaib Azam: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Farzeen Munir:** Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Investigation, Data curation, Conceptualization. **Ville Kyrki:** Writing – review & editing, Supervision, Resources, Project administration, Investigation. **Tomasz Piotr Kucner:** Supervision, Validation, Writing – review & editing. **Moongu Jeon:** Writing – review & editing, Supervision, Resources, Project administration, Investigation, Funding acquisition. **Witold Pedrycz:** Writing – review & editing, Supervision, Resources, Investigation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data is public available: nuScene and Carla datasets.

Acknowledgments

This work is supported by the Academy of Finland Flagship program: Finnish Center for Artificial Intelligence (FCAI), and also by GIST-MIT Collaboration Research Program.

Table 10

Impact of Backbone Networks and Attention Layers on L2 Distance Error: The comparative analysis presents L2 distance errors at 1 s, 2 s, and 3 s time horizons, along with the average, for the proposed method, and its early and late fusion variants. Different configurations of backbone networks (ResNet18, ResNet34, ResNet50) and attention layers (2, 6, 8, 12) were evaluated to determine their effect on the accuracy of waypoint prediction.

Network Parameters	Value	L2 (m)											
		1 s	2 s	3 s	Avg	1 s	2 s	3 s	Avg	1 s	2 s	3 s	Avg
		(Proposed Method)				(Early Fusion)				(Late Fusion)			
Backbone Network	ResNet18	0.49	0.72	1.23	0.81	1.68	2.47	3.21	2.45	0.54	0.91	1.31	0.92
	ResNet34	0.41	0.66	1.12	0.73	1.62	2.33	3.10	2.35	0.46	0.82	1.23	0.84
	ResNet50	**0.35**	**0.61**	**1.01**	**0.66**	**1.55**	**2.25**	**2.99**	**2.26**	**0.42**	**0.74**	**1.15**	**0.77**
Attention Layers	2	0.44	0.70	1.16	0.77	1.67	2.41	3.18	2.42	0.51	0.82	1.27	0.86
	6	0.40	0.66	1.11	0.72	1.62	2.35	3.09	2.35	0.45	0.79	1.22	0.82
	8	0.38	0.63	1.04	0.68	1.57	2.29	3.03	2.29	0.44	0.77	1.18	0.80
	12	**0.35**	**0.61**	**1.01**	**0.66**	**1.55**	**2.25**	**2.99**	**2.26**	**0.42**	**0.74**	**1.15**	**0.77**

Table 11

Evaluation of Backbone Networks and Attention Layers on Town05 Long Benchmark: This quantitative analysis showcases the driving score (DS), route completion (RC), and infraction score (IS) metrics for the proposed method and the early and late fusion variants. The evaluations are conducted using different backbone networks (ResNet18, ResNet34, ResNet50) and varying numbers of attention layers (2, 6, 8, 12) to assess their influence on the performance of each method.

Network Parameters	Value	Metrics								
		DS ↑	RC ↑	IS ↑	DS ↑	RC ↑	IS ↑	DS ↑	RC ↑	IS ↑
		(Proposed Method)			(Early Fusion)			(Late Fusion)		
Backbone Network	ResNet18	58.21	87.35	0.65	46.52	72.56	0.59	52.78	80.15	0.62
	ResNet34	65.47	91.56	0.69	49.39	78.47	0.64	57.25	85.47	0.68
	ResNet50	**68.30**	**96.50**	**0.75**	**52.74**	**80.12**	**0.65**	**60.35**	**87.45**	**0.72**
Attention Layers	2	60.55	89.15	0.63	40.63	69.18	0.55	51.85	79.55	0.65
	6	65.74	92.54	0.68	45.87	73.62	0.59	55.73	81.36	0.67
	8	66.45	96.15	0.71	47.31	76.54	0.63	59.68	85.43	0.70
	12	**68.30**	**96.50**	**0.75**	**52.74**	**80.12**	**0.65**	**60.35**	**87.45**	**0.72**

Table 12

Evaluation of Backbone Networks and Attention Layers on Longest6 Benchmark: This quantitative results showcases different backbone networks (ResNet18, ResNet34, ResNet50) and varying numbers of attention layers (2, 6, 8, 12) to assess their influence on the performance of each method.

Network Parameters	Value	Metrics								
		DS ↑	RC ↑	IS ↑	DS ↑	RC ↑	IS ↑	DS ↑	RC ↑	IS ↑
		(Proposed Method)			(Early Fusion)			(Late Fusion)		
Backbone Network	ResNet18	57.45	74.23	0.71	41.20	58.63	0.58	50.95	64.37	0.71
	ResNet34	63.32	77.56	0.78	46.47	61.35	0.63	55.49	69.54	0.74
	ResNet50	**67.43**	**80.54**	**0.81**	**47.35**	**65.50**	**0.69**	**59.15**	**71.73**	**0.77**
Attention Layers	2	59.31	70.10	0.69	40.00	58.43	0.59	48.96	63.54	0.68
	6	62.67	73.21	0.73	42.71	61.53	0.62	51.47	66.19	0.71
	8	64.15	78.95	0.79	45.17	64.10	0.67	56.78	69.27	0.74
	12	**67.43**	**80.54**	**0.81**	**47.35**	**65.50**	**0.69**	**59.15**	**71.73**	**0.77**

References

Azam, S., Munir, F., Sheri, A.M., Kim, J., Jeon, M., 2020. System, design and experimental validation of autonomous vehicle in an unconstrained environment. Sensors 20 (21), 5999.

Behl, A., Chitta, K., Prakash, A., Ohn-Bar, E., Geiger, A., 2020. Label efficient visual abstractions for autonomous driving. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS, IEEE, pp. 2338–2345.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S., 2020. End-to-end object detection with transformers. In: European Conference on Computer Vision. Springer, pp. 213–229.

Chen, D., Koltun, V., Krähenbühl, P., 2021. Learning to drive from a world on rails. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15590–15599.

Chen, D., Krähenbühl, P., 2022. Learning from all vehicles. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17222–17231.

Chen, X., Ma, H., Wan, J., Li, B., Xia, T., 2017. Multi-view 3d object detection network for autonomous driving. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1907–1915.

Chen, K., Oldja, R., Smolyanskiy, N., Birchfield, S., Popov, A., Wehr, D., Eden, I., Pehserl, J., 2020a. Mvlidarnet: Real-time multi-class scene understanding for autonomous driving using multiple views. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS, IEEE, pp. 2288–2294.

Chen, D., Zhou, B., Koltun, V., Krähenbühl, P., 2020b. Learning by cheating. In: Conference on Robot Learning. PMLR, pp. 66–75.

Chitta, K., Prakash, A., Geiger, A., 2021. Neat: Neural attention fields for end-to-end autonomous driving. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15793–15803.

Codevilla, F., Müller, M., López, A., Koltun, V., Dosovitskiy, A., 2018. End-to-end driving via conditional imitation learning. In: 2018 IEEE International Conference on Robotics and Automation. ICRA, IEEE, pp. 4693–4700.

Codevilla, F., Santana, E., López, A.M., Gaidon, A., 2019. Exploring the limitations of behavior cloning for autonomous driving. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9329–9338.

Dong, J., Chen, S., Zong, S., Chen, T., Labi, S., 2021. Image transformer for explainable autonomous driving system. In: 2021 IEEE International Intelligent Transportation Systems Conference. ITSC, IEEE, pp. 2732–2737.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al., 2020. An image is worth 16 × 16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929.

Fadadu, S., Pandey, S., Hegde, D., Shi, Y., Chou, F.C., Djuric, N., Vallespi-Gonzalez, C., 2022. Multi-view fusion of sensor data for improved perception and prediction in autonomous driving. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 2349–2357.

Haris, M., Glowacz, A., 2022. Navigating an automated driving vehicle via the early fusion of multi-modality. Sensors 22 (4), 1425.

Hu, S., Chen, L., Wu, P., Li, H., Yan, J., Tao, D., 2022. St-p3: End-to-end vision-based autonomous driving via spatial-temporal feature learning. In: European Conference on Computer Vision. Springer, pp. 533–549.

Hu, P., Huang, A., Dolan, J., Held, D., Ramanan, D., 2021. Safe local motion planning with self-supervised freespace forecasting. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, pp. 12732–12741.

Hu, Y., Yang, J., Chen, L., Li, K., Sima, C., Zhu, X., Chai, S., Du, S., Lin, T., Wang, W., et al., 2023. Planning-oriented autonomous driving. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17853–17862.

Huang, Z., Lv, C., Xing, Y., Wu, J., 2020. Multi-modal sensor fusion-based deep neural network for end-to-end autonomous driving with scene understanding. IEEE Sens. J. 21 (10), 11781–11790.

Huang, Z., Mo, X., Lv, C., 2022. Multi-modal motion prediction with transformer-based neural network for autonomous driving. In: 2022 International Conference on Robotics and Automation. ICRA, IEEE, pp. 2605–2611.

Huang, Z., Sun, S., Zhao, J., Mao, L., 2023. Multi-modal policy fusion for end-to-end autonomous driving. Inf. Fusion 98, 101834.

Jia, X., Wu, P., Chen, L., Xie, J., He, C., Yan, J., Li, H., 2023. Think twice before driving: Towards scalable decoders for end-to-end autonomous driving. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21983–21994.

Jiang, B., Chen, S., Xu, Q., Liao, B., Chen, J., Zhou, H., Zhang, Q., Liu, W., Huang, C., Wang, X., 2023. VAD: Vectorized scene representation for efficient autonomous driving. arXiv preprint arXiv:2303.12077.

Khan, M.A., Sayed, H.E., Malik, S., Zia, T., Khan, J., Alkaabi, N., Ignatious, H., 2022. Level-5 autonomous driving—are we there yet? A review of research literature. ACM Comput. Surv. 55 (2), 1–38.

Li, Z., Wang, W., Li, H., Xie, E., Sima, C., Lu, T., Qiao, Y., Dai, J., 2022. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In: European Conference on Computer Vision. Springer, pp. 1–18.

Li, L.L., Yang, B., Liang, M., Zeng, W., Ren, M., Segal, S., Urtasun, R., 2020. End-to-end contextual perception and prediction with interaction transformer. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS, IEEE, pp. 5784–5791.

Liang, M., Yang, B., Wang, S., Urtasun, R., 2018. Deep continuous fusion for multi-sensor 3d object detection. In: Proceedings of the European Conference on Computer Vision. ECCV, pp. 641–656.

Liu, Z., Tang, H., Amini, A., Yang, X., Mao, H., Rus, D.L., Han, S., 2023. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In: 2023 IEEE International Conference on Robotics and Automation. ICRA, IEEE, pp. 2774–2781.

Macaluso, E., 2006. Multisensory processing in sensory-specific cortical areas. Neuroscientist 12 (4), 327–338.

Man, Y., Gui, L.Y., Wang, Y.X., 2023. BEV-guided multi-modality fusion for driving perception. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21960–21969.

Meyer, G.P., Charland, J., Pandey, S., Laddha, A., Gautam, S., Vallespi-Gonzalez, C., Wellington, C.K., 2020. Laserflow: Efficient and probabilistic object detection and motion forecasting. IEEE Robot. Autom. Lett. 6 (2), 526–533.

Munir, F., Azam, S., Yow, K.C., Lee, B.G., Jeon, M., 2023. Multimodal fusion for sensorimotor control in steering angle prediction. Eng. Appl. Artif. Intell. 126, 107087.

Natan, O., Miura, J., 2022. End-to-end autonomous driving with semantic depth cloud mapping and multi-agent. IEEE Trans. Intell. Veh. 8 (1), 557–571.

Philion, J., Fidler, S., 2020. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16. Springer, pp. 194–210.

Prakash, A., Chitta, K., Geiger, A., 2021. Multi-modal fusion transformer for end-to-end autonomous driving. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7077–7087.

Schroeder, C.E., Foxe, J., 2005. Multisensory contributions to low-level,'unisensory'processing. Curr. Opin. Neurobiol. 15 (4), 454–458.

Schwarting, W., Alonso-Mora, J., Rus, D., 2018. Planning and decision-making for autonomous vehicles. Annu. Rev. Control Robot. Auton. Syst. 1, 187–210.

Shao, H., Wang, L., Chen, R., Li, H., Liu, Y., 2022. Safety-enhanced autonomous driving using interpretable sensor fusion transformer. arXiv preprint arXiv:2207.14024.

Shao, H., Wang, L., Chen, R., Li, H., Liu, Y., 2023. Safety-enhanced autonomous driving using interpretable sensor fusion transformer. In: Conference on Robot Learning. PMLR, pp. 726–737.

Singh, A., 2023. Transformer-based sensor fusion for autonomous driving: A survey. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3312–3317.

Sobh, I., Amin, L., Abdelkarim, S., Elmadawy, K., Saeed, M., Abdeltawab, O., Gamal, M., El Sallab, A., 2018. End-to-end multi-modal sensors fusion system for urban automated driving.

Tang, Q., Liang, J., Zhu, F., 2023. A comparative review on multi-modal sensors fusion based on deep learning. Signal Process. 109165.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I., 2017. Attention is all you need. Adv. Neural Inf. Process. Syst. 30.

Vora, S., Lang, A.H., Helou, B., Beijbom, O., 2020. Pointpainting: Sequential fusion for 3d object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4604–4612.

Wu, P., Jia, X., Chen, L., Yan, J., Li, H., Qiao, Y., 2022. Trajectory-guided control prediction for end-to-end autonomous driving: A simple yet strong baseline. Adv. Neural Inf. Process. Syst. 35, 6119–6132.

Xiao, Y., Codevilla, F., Gurram, A., Urfalioglu, O., López, A.M., 2020. Multimodal end-to-end autonomous driving. IEEE Trans. Intell. Transp. Syst. 23 (1), 537–547.

Yao, Z., Gholami, A., Shen, S., Mustafa, M., Keutzer, K., Mahoney, M., 2021. Adahessian: An adaptive second order optimizer for machine learning. In: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, No. 12. pp. 10665–10673.

Ye, T., Jing, W., Hu, C., Huang, S., Gao, L., Li, F., Wang, J., Guo, K., Xiao, W., Mao, W., et al., 2023. Fusionad: Multi-modality fusion for prediction and planning tasks of autonomous driving. arXiv preprint arXiv:2308.01006.

Yurtsever, E., Lambert, J., Carballo, A., Takeda, K., 2020. A survey of autonomous driving: Common practices and emerging technologies. IEEE Access 8, 58443–58469.

Zeng, W., Luo, W., Suo, S., Sadat, A., Yang, B., Casas, S., Urtasun, R., 2019. End-to-end interpretable neural motion planner. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8660–8669.

Zhang, J., Huang, Z., Ohn-Bar, E., 2023. Coaching a teachable student. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7805–7815.

Zhang, Z., Liniger, A., Dai, D., Yu, F., Van Gool, L., 2021. End-to-end urban driving by imitating a reinforcement learning coach. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15222–15232.

Zhou, B., Krähenbühl, P., 2022. Cross-view transformers for real-time map-view semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13760–13769.

Zhou, B., Krähenbühl, P., Koltun, V., 2019. Does computer vision matter for action? Science Robotics 4 (30), eaaw6661.