Safety-Enhanced Autonomous Driving Using Interpretable Sensor Fusion Transformer

Anonymous Author(s)
Affiliation
Address
email

Abstract: Large-scale deployment of autonomous vehicles has been continually delayed due to safety concerns. On the one hand, comprehensive scene understanding is indispensable, a lack of which would result in vulnerability to rare but complex traffic situations, such as the sudden emergence of unknown objects. However, reasoning from a global context requires access to sensors of multiple types and adequate fusion of multi-modal sensor signals, which is difficult to achieve. On the other hand, the lack of interpretability in learning models also hampers the safety with unverifiable failure causes. In this paper, we propose a safety-enhanced autonomous driving framework, named Interpretable Sensor Fusion Transformer (InterFuser), to fully process and fuse information from multi-modal multi-view sensors for achieving comprehensive scene understanding and adversarial event detection. Besides, intermediate interpretable features are generated from our framework, which provide more semantics and are exploited to better constrain actions to be within the safe sets. We conducted extensive experiments on CARLA benchmarks, where our model outperforms prior methods, ranking the first on the public CARLA Leaderboard.

Keywords: Autonomous driving, sensor fusion, transformer, safety

1 Introduction

Recently, rapid progress has been witnessed in the field of autonomous driving, while the scalable and practical deployment of autonomous vehicles on public roads is still far from feasible. Their incompetence is mainly observed in high-traffic-density scenes, where a large number of obstacles and dynamic objects are involved in the decision making. In these cases, currently deployed systems could exhibit incorrect or unexpected behaviours leading to catastrophic accidents [1, 2]. While many factors contribute to such safety concerns, two of the major challenges are: 1) how to recognize rare adverse events of long-tail distributions, such as the sudden emergence of pedestrians from road sides, and vehicles running a red light, which require a better understanding of the scenes with multi-modal multi-view sensor inputs; 2) how to verify the decision-making process, in other words, identify functioning/malfunctioning conditions of the system and the causes for failures, which requires interpretability of the decision-making system.

Safe and reliable driving necessitates comprehensive scene understanding. However, a single sensor generally cannot provide adequate information for perceiving the driving scenes. Single-image approaches can hardly capture the surrounding environment from multiple perspectives and cannot provide 3D measurements of the scene, while single-LiDAR approaches cannot take semantic information such as traffic lights into account. Though there are existing works fusing information from multiple sensors, they either match geometric features between image space and LiDAR projection space by locality assumption [3, 4], or simply concatenate multi-sensor features [5, 6]. The interactions and relationships between multi-modal features are seldomly modeled, such as the interactions between multiple dynamic agents and traffic lights, or features in different views and

Submitted to the 6th Conference on Robot Learning (CoRL 2022). Do not distribute.

modalities. To encourage reasoning with a global context, the attention mechanism of Transformer [7] is utilized. The recent TransFuser [8] adopts internal feature propagation and aggregation via a multi-stage CNN-transformer architecture to fuse bi-modal inputs, which harms sensor scalability and is limited to fusion between LiDAR and a single-view image. In this paper, we take a one-stage architecture to effectively fuse information from multi-modal multi-view sensors and achieve significant improvement. As shown in Fig. 1, we consider LiDAR input and multi-view images (left, front, right, and focus) as complementary inputs to achieve comprehensive full-scene understanding.

On the other hand, existing end-to-end driving methods barely have a safety ensurance mechanism due to the lack of interpretability of how the control signal is generated. To tackle such a challenge, there are efforts to verify the functioning conditions of neural networks instead of directly understanding the models [9, 10, 11]. Though being helpful for choosing different models for different conditions, these methods still lack feedback from the failure causes for further improvement. Inspired by humans' information collecting process [12], in addition to generating actions, our model also outputs intermediate interpretable features, which we call *safety mind map*. As shown in Fig. 1, the safety mind map provides information on surrounding objects and traffic signs. Unveiling the process of perceiving and decision-making, our model is improvable with clear failure conditions and causes. Moreover, by exploiting this intermediate interpretable information as a safety constraint heuristic, we can constrain the actions within a safe action set to further enhance driving safety.

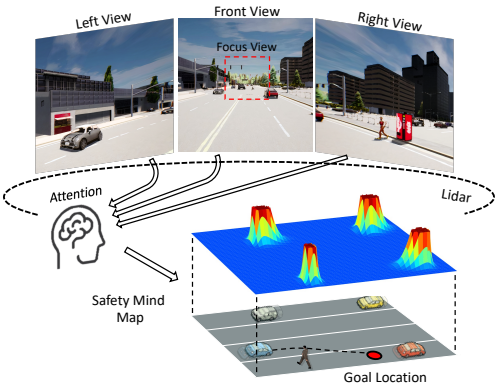


Figure 1: Safe and efficient driving requires comprehensive scene understanding by fusing information from multiple sensors. Peeking into the intermediate interpretable features of learning models can also unveil the model's decision basis. Such features enable improvable systems with access to failure causes, and can be used as safety heuristic to constrain actions within the safe set.

In this paper, we propose a safety-enhanced driving framework called Interpretable Sensor Fusion Transformer (InterFuser), in which information from multi-modal multi-view sensors is fused, and driving safety is enhanced by providing intermediate interpretable features as safety constraint heuristics. Our contributions are three-fold:

1. We propose a novel Interpretable Sensor Fusion Transformer (InterFuser) to encourage global contextual perception and reasoning in different modalities and views.
2. Our framework enhances the safety and interpretability of end-to-end driving by outputting intermediate features of the model and constraining actions within safe sets.
3. We experimentally validated our method on several CARLA benchmarks with complex and adversarial urban scenarios. Our model outperformed prior methods, ranking the first on the public CARLA Leaderboard.

2 Related work

End-to-end autonomous driving in urban scenarios The research of end-to-end auto-driving based on the simulator of urban scenarios has become the mainstream. The topic starts from the development of an urban driving simulator: CARLA [13], together with Conditional Imitation Learning (CIL) [3] proposed. A series of work follow this way, yielding conditional end-to-end driving [14, 15] in urban scenarios. LBC [16] proposed the mimicking methods aimed at training image-input networks with the supervision of a privileged model or squeezed model. Chitta et al. [17] presents neural attention fields to enable the reasoning for end-to-end driving models. Imitation learning(IL) approaches lack interpretability and their performance is limited by their handcrafted expert autopilot. Hence,

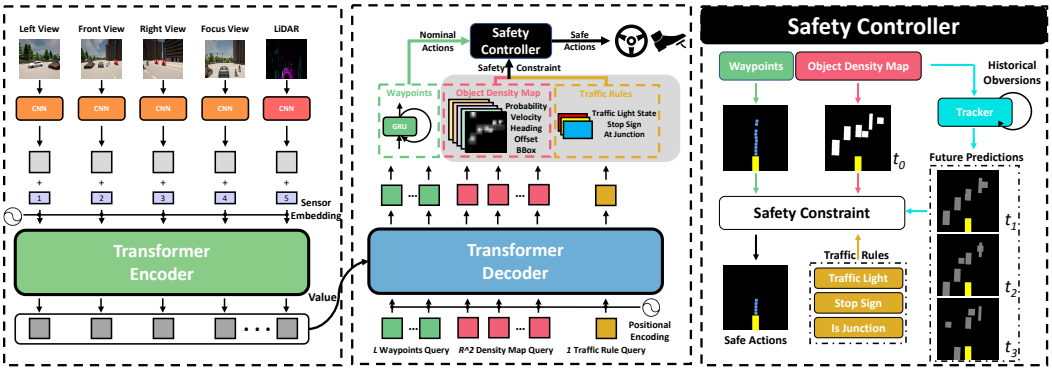


Figure 2: Overview of our approach. We first use CNN backbones to extract features from multi-modal multi-view sensor inputs. The tokens from different sensors are then fused in the transformer encoder. Three types of queries are then fed into the transformer decoder to predict waypoints, object density maps and traffic rules respectively. At last, by recovering the traffic scene from the predicted object density map and utilizing the tracker to get the future predictions of other objects, a safety controller is applied to ensure safe and efficient driving in complex traffic situations.

researchers develop Reinforcement Learning (RL) agents to interact with simulated environments. Latent DRL [18] generates intermediate feature embedding from a top-down view image by training a variational auto-encoder. With the aforementioned mimicking tricks, Roach [19] trained an RL-based privileged model as the expert agent to provide demonstrations for the IL agent. Toromanoff et al. [6] proposes to use hidden states which is supervised by semantic information as the input of RL policy.

Transformer model in vision comprehension Transformer was originally established in Natural Language Processing (NLP) [7]. The attention mechanism demonstrates to be a powerful module in image processing tasks. Vision Transformer (ViT) [20] computes relationships among pixel blocks with a reasonable computation cost. Later generations move on to generalize Transformer to other computer vision tasks [21, 22, 23, 24] or digging deeper to perform better [25, 26]. The attention mechanism brings a new entry point for modality fusion. TransformerFusion [27] reconstruct 3D scenes that takes monocular video as input with a transformer architecture. TransFuser [8] exploits several transformer modules for the fusion of intermediate features of front view and LiDAR. However, such a sensor-pair intense fusion approach hardly scales to more sensors, while information from side views like an emerging pedestrian, and a focus view like the traffic light, are critical for scene understanding and safe driving.

Safe and interpretable driving Safety has long been studied in the traditional control community. However, in autonomous driving, uncertain behaviors [28] and diverse driving preferences of drivers [29] deteriorate the safety concern. Some works take the uncertain and heterogeneous nature of humans into the formulation of control methods for safety assurance [30, 31]. However, these rule-based methods hardly scale to complex environments due to the extensive human labor required. When learning-based methods arise, their lack of interpretability becomes a new puzzle in the way. There are efforts taking a bypass, verifying the functioning conditions of neural network models instead of directly understanding them [9, 10, 11]. However, feedback on the failure causes and solutions are still wanted. Recently, some works design auxiliary tasks to output interpretable semantic features [32, 33], which is showing a great improvement on both the performance and interpretability. In this paper, we output the intermediate interpretable features for more semantics, which effectively enhances the driving safety.

3 Method

As illustrated in Fig. 2, the proposed framework consists of three main components: 1) a multi-view multi-modal fusion transformer encoder that integrates the signals from multiple RGB cameras and LiDAR; 2) a transformer decoder providing the low-level actions, and interpretable intermediate

features including the ego vehicle's future trajectory, the object density map, and the traffic rule signals; 3) a safety controller which utilizes the interpretable intermediate features to constrain the low-level control within the safe set. This section will introduce the input/output representation, model architecture and safety controller.

3.1 Input and Output Representations

Input representations Four sensors are utilized, including three RGB cameras (left, front and right) and one LiDAR sensor. Four image inputs are obtained from the three cameras. In addition to the left, front and right image input $\{\mathbf{I}_{\text{left}}, \mathbf{I}_{\text{front}}, \mathbf{I}_{\text{right}}\}$, we also design a focusing-view image input $\mathbf{I}_{\text{focus}}$ To capture the status of distant traffic lights by cropping the center patch of the raw front RGB image. For the LiDAR point clouds, we follow previous works [34, 35, 8] to convert the LiDAR point cloud data into a 2-bin histogram over a 2-dimensional Bird's Eye View (BEV) grid, resulting in a two-channel LiDAR bird-view projection image input $\mathbf{I}_{\text{lidar}}$.

Output representations Our model generates two types of outputs: safety-insensitive and safety-sensitive outputs. For safety-insensitive outputs, InterFuser predicts a path with $L = 10$ waypoints for the ego vehicle to steer toward. It guides the future driving route of the ego vehicle. However, driving along the path without a proper speed might be unsafe and violate actual traffic rules. Therefore, InterFuser additionally predicts the safety-sensitive outputs, consisting of an object density map and traffic rule information. The object density map $M \in \mathbb{R}^{R\times R\times 7}$ provides 7 features for detected objects, such as vehicles, pedestrians and bicycles. $M_{i,j}$ indicates an $1m \times 1m$ grid area indexed by spatial coordinates (i, j) where the ego vehicle is taken as the origin and the y-axis is the forward direction. So the map covers R meters in front of the ego vehicle and $\frac{R}{2}$ meters on its two sides. The 7 channels of the map are the probability of the existence of an object, 2-dimensional offset from the center of the $1m \times 1m$ grid, the size of the object bounding box, the object heading, and the object velocity. Besides, InterFuser also outputs the traffic rule information, including the traffic light state, whether there is a stop sign ahead, and whether the ego vehicle is at an intersection.

3.2 Model architecture

Backbone For each image input and LiDAR input $I \in \mathbb{R}^{3\times H_0\times W_0}$, we use a conventional CNN backbone ResNet [36] to generate a lower-resolution feature map $\mathbf{f} \in \mathbb{R}^{C\times H\times W}$. We set $C = 2048$ and $(H, W) = (\frac{H_0}{32}, \frac{W_0}{32})$ in experiments.

Transformer encoder For feature map f of each sensor input, we first take a 1×1 convolution to obtain a lower-channel feature map $\mathbf{z} \in \mathbb{R}^{d\times H\times W}$. The spatial dimension of each feature map $\mathbf{z}$ is then collapsed into one dimension, resulting in $d \times HW$ tokens. Fixed 2D sinusoidal positional encoding $\mathbf{e} \in \mathbb{R}^{d\times HW}$ is added to each token to retain positional information in each sensor input, and learnable sensor embedding $\mathbf{s} \in \mathbb{R}^{d\times N}$ is added to distinguish tokens from N different sensors:

$$\mathbf{v}_i^{(x,y)} = \mathbf{z}_i^{(x,y)} + \mathbf{s}_i + \mathbf{e}^{(x,y)} \tag{1}$$

Where $\mathbf{z}_i$ represents the tokens extracted from i-th view, x, y denotes the token's coordinate index in that sensor. Finally we concatenate the tokens from all sensors, which are then passed through a transformer encoder with K standard transformer layers. Each layer $\mathcal{K}$ consists of Multi-Headed Self-Attention [7], MLP blocks and layer normalisation [37].

Transformer decoder The decoder follows standard transformer architecture, transforming some query embeddings of size d using K layers of multi-headed self-attention mechanisms. Three types of queries are designed: L waypoints queries, R^2 density map queries and one traffic rule query. In each decoder layer, we employ these queries to inquire about the spatial information from the multi-modal multi-view features via the attention mechanism. Since the transformer decoder is also permutation-invariant, the above query embeddings are the same for the decoder and can not produce different results. To this end, we add learnable positional embeddings to these query embeddings. The results of these queries are then independently decoded into L waypoints, one density map and the traffic rule by the following prediction headers.

Prediction headers The transformer decoder is followed by three parallel prediction modules to predict the waypoints, the object density map and the traffic rule respectively. For waypoints prediction, following [38, 10], we take a single layer GRU [39] to auto-regressively predict a sequence of L future waypoints $\{\mathbf{w}_l\}_{l=1}^{L}$. The GRU predicts the t-th waypoint by taking in the hidden state from step $t-1$ and the t-th waypoint embedding from the transformer decoder. Note that we first predict each step's differential displacement and then recover the exact position by accumulation. To inform the waypoints predictor of the ego vehicle's goal location, we initialize the initial hidden state of GRU with a 64-dimensional vector embedded by the GPS coordinates of the goal location with a linear projection layer. For the density map prediction, the corresponding R^2 d-dimensional embeddings from the transformer decoder are passed through a 3-layer MLP with a ReLU activation function to get a $R^2 \times 7$ feature map. We then reshape it into $\mathbf{M} \in \mathbb{R}^{R \times R \times 7}$ to obtain the object density map. For traffic rule prediction, the corresponding embedding from the transformer decoder is passed through a single linear layer to predict the state of traffic light ahead, whether there is a stop sign ahead and whether the ego vehicle is at an intersection.

Loss Function The loss function is designed to encourage predicting the desired waypoints ($\mathcal{L}_{pt}$), object density map ($\mathcal{L}_{map}$), and traffic information ($\mathcal{L}_{tf}$):

$$\mathcal{L} = \lambda_{pt}\mathcal{L}_{\text{pt}} + \lambda_{map}\mathcal{L}_{map} + \lambda_{tf}\mathcal{L}_{tf}, \tag{2}$$

where λ balances the three loss terms. Readers can refer to Appendix A for detailed description.

3.3 Safety Controller

With the waypoints and intermediate interpretable features (object density map and traffic rule) output from the transformer decoder, we are able to constrain the actions into the safe set. Specifically, we use PID controllers to get two low-level actions. The lateral steering action aligns the ego vehicle to the desired heading ψ_d, which is simply the average heading of the first two waypoints. The longitudinal acceleration action aims to catch the desired speed v_d. The determination of v_d needs to take surrounding objects into account to ensure safety, for which we resort to the object density map.

The object in a grid of the object density map $M \in \mathbb{R}^{R \times R \times 7}$ is described by an object existence probability, a 2-dimensional offset from the grid center, a 2-dimensional bounding box and a heading angle. We recognize the existence of an object in a grid once one of the following criteria is met: 1) if the object's existence probability in the grid is higher than threshold$_1$. 2) if the object existence probability in the grid is the local maximum in surrounding grids and greater than threshold$_2$ (threshold$_2$ $<$ threshold$_1$). While the first rule is intuitive, the second rule is set to recognize the existence of an object with high position uncertainty. In addition to the current state of objects, the safety controller also needs to consider their future trajectory. We first design a tracker to monitor and record their historical dynamics. We then predict their future trajectory by propagating their historical dynamics forward in time with moving average.

With the recovered surrounding scene and future forecasting of these objects, we then can obtain s_t, the maximum safe distance the ego-vehicle can drive at time step t. The desired velocity v_d with enhanced safety is then derived by solving a linear programming problem. Note that to avoid attractions of unsafe sets and future safety intractability, we also augment the shape of objects, and consider the actuation limit and ego vehicle's dynamic constraint. For details of desired velocity v_d optimization, please refer to Appendix B. Besides the object density map, the predicted traffic rule is also utilized for safe driving. The ego-vehicle will perform an emergency stop if the traffic light is not green or there is a a stop sign ahead.

Note that while more advanced trajectory prediction methods [40, 41, 42] and safety controller [30, 43] can be used, we empirically found our dynamics propagation with moving average and linear programming controller sufficient. In case of more complicated driving tasks, those advanced algorithms can be easily integrated into our framework in the future.

4 Experiments

4.1 Experiment Setup

We implemented our approach on the open-source CARLA simulator with version 0.9.10.1, including 8 towns and 21 kinds of weather. Please refer to Appendix C for implementation details.

Dataset collection We ran a rule-based expert agent on all 8 kinds of towns and weather with 2 FPS, to collect an expert dataset of 3M frames (410 hours). Note that the expert agent has the access to the privileged information in the CARLA simulator. For the diversity of the dataset, we randomly generated different routes, spawn dynamic objects and adversarial scenarios.

Benchmark We evaluated our methods on three benchmarks: CARLA public leaderboard [44], the Town05 benchmark [8] and CARLA 42 Routes benchmark [17]. In these benchmarks, the ego vehicle is required to navigate along predefined routes without collision or violating traffic rules in existence of adversarial events[1]. At each run, the benchmark randomly spawns start/destination points, and generates a sequence of sparse goal locations in GPS coordinates. Our method uses these sparse goal locations to guide the driving without manually setting discrete navigational commands (go straight, lane changing, turning). Please refer to Appendix D for detailed descriptions of the two benchmarks.

Metrics Three metrics introduced by the CARLA LeaderBoard are used for evaluation: the route completion ratio (RC), infraction score (IS), and the driving score (DS). Route completion is the percentage of the route completed. The infraction score is a performance discount value. When the ego-vehicle commits an infraction or violates a traffic rule, the infractions score will decay by a corresponding percentage. The driving score is the product of the route completion ratio and the infraction score, and thus is a more comprehensive metric to describe both driving progress and safety.

4.2 Comparison to the state of the art

Table 1 compares our method with top 10 state-of-the-art methods on the CARLA leaderboard [44]. TCP is an anonymous submission without reference. LAV [45] trains the driving agent with the dataset collected from all the vehicles that it observes. Transfuser [8, 46] is an imitation learning method where the agent uses transformer to fuse information from the front camera image and LiDAR information. The entries "Latent Transfuser" and "Transfuser+" are variants of Transfuser. GRIAD [47] proposes General Reinforced Imitation which combines benefits from exploration and expert data. Rails [49] uses a tabular dynamic-programming evaluation to compute action-values. IARL [6] is a method based on a model-free reinforcement-learning. NEAT [17] proposes neural attention fields which enables the reasoning for end-to-end imitation learning.

Rank	Method	Driving Score	Route Completion	Infraction Score
1	InterFuser (ours)	**76.18**	88.23	0.84
2	TCP	75.14	85.63	**0.87**
3	LAV [45]	61.85	**94.46**	0.64
4	TransFuser [46]	61.18	86.69	0.71
5	Latent TransFuser [46]	45.20	66.31	0.72
6	GRIAD [47]	36.79	61.85	0.60
7	TransFuser+ [48]	34.58	69.84	0.56
8	Rails [49]	31.37	57.65	0.56
9	IARL [6]	24.98	46.97	0.52
10	NEAT [17]	21.83	41.71	0.65

Table 1: Performance comparison on the public CARLA leaderboard [44] (accessed Jun 2022). All three metrics are higher the better. Our Interfuser ranks first on the leaderboard, with the highest driving score, the second highest route completion, and the second highest infraction score.

Our InterFuser ranks first on the leaderboard, with the highest driving score 76.18, the second highest route completion 88.23, and the second highest infraction score 0.84. We also compared our method with other methods on the Town05 benchmark and CARLA 42 Routes benchmark. As shown in Table 5 and Table 6 of the appendix, our method beats other methods.

[1]The adversarial events include bad road conditions, front vehicle's sudden brake, unexpected entities rushing into the road from occluded regions, vehicles running a red traffic light, etc. Please refer to https://leaderboard.carla.org/scenarios/ for detailed descriptions of adversarial events.

	Driving score	Route compl.	Infrac. score	Collision static	Collision pedestrian	Collision vehicle	Red light infraction	Agent blocked
	%, ↑	%, ↑	%, ↑	#/Km, ↓	#/Km, ↓	#/Km, ↓	#/Km, ↓	#/Km, ↓
F	40.3 ± 4.3	40.3 ± 5.2	$\mathbf{99.9} \pm 0.0$	0.09 ± 0.01	0.04 ± 0.01	0.11 ± 0.02	0.09 ± 0.02	$\mathbf{0} \pm 0$
$F + Li$	47.2 ± 0.9	48.9 ± 2.9	96.6 ± 1.1	0.06 ± 0.01	0.02 ± 0.02	$\mathbf{0.09} \pm 0.02$	0.10 ± 0.02	$\mathbf{0} \pm 0$
$F + LR$	43.0 ± 5.4	43.2 ± 6.2	98.0 ± 2.7	0.05 ± 0.01	0.01 ± 0	$\mathbf{0.09} \pm 0.03$	0.12 ± 0.03	$\mathbf{0} \pm 0$
$F + LR + Fc$	46.4 ± 3.1	48.5 ± 3.3	97.1 ± 5.0	0.07 ± 0.01	0.01 ± 0.01	0.1 ± 0.02	0.04 ± 0.02	0.02 ± 0.02
$F + LR + Li$	49.2 ± 0.7	55.3 ± 2.8	89.6 ± 0.7	$\mathbf{0.01} \pm 0.01$	$\mathbf{0} \pm 0$	$\mathbf{0.09} \pm 0.05$	0.10 ± 0.02	0.04 ± 0.02
$F + LR + Fc + Li$	$\mathbf{51.6} \pm 3.4$	$\mathbf{58.6} \pm 5.2$	88.9 ± 2.5	$\mathbf{0.01} \pm 0.01$	$\mathbf{0} \pm 0$	$\mathbf{0.09} \pm 0.05$	$\mathbf{0.02} \pm 0.02$	0.07 ± 0.01

Table 2: Ablation study for different sensor inputs. F, LR, Fc, Li denotes the front view, left and right view, focusing view, LiDAR BEV representations respectively. Our method performs the best when all sensor inputs are leveraged.

	Driving score	Route compl.	Infrac. score	Collision static	Collision pedestrian	Collision vehicle	Red light infraction	Agent blocked
	%, ↑	%, ↑	%, ↑	#/Km, ↓	#/Km, ↓	#/Km, ↓	#/Km, ↓	#/Km, ↓
InterFuser (ours)	$\mathbf{51.6} \pm 3.4$	$\mathbf{58.6} \pm 5.2$	88.9 ± 2.5	$\mathbf{0.01} \pm 0.01$	$\mathbf{0} \pm 0$	$\mathbf{0.09} \pm 0.05$	0.10 ± 0.02	0.07 ± 0.01
No sensr/pos embd	48.6 ± 4.7	52.6 ± 6.1	91.7 ± 2.4	0.05 ± 0.04	0.01 ± 0.01	0.11 ± 0.10	0.05 ± 0.01	0.08 ± 0.1
Concatenated input	21.5 ± 7.6	21.5 ± 8.6	$\mathbf{97.2} \pm 3.9$	0.06 ± 0	0.03 ± 0.02	0.26 ± 0.05	0.20 ± 0.02	$\mathbf{0} \pm 0.01$
No attn btw views	46.8 ± 6.6	49.6 ± 9.8	94.1 ± 3.5	0.07 ± 0.01	0.01 ± 0.01	0.1 ± 0.02	0.04 ± 0.02	0.01 ± 0.01
No safety contrl	43.9 ± 4.7	45.6 ± 5.9	96.3 ± 5.2	0.05 ± 0.03	0.02 ± 0.01	0.13 ± 0.10	$\mathbf{0.02} \pm 0.02$	0.08 ± 0.2

Table 3: Ablation study for sensor/position embedding (No sensr/pos embd), sensor fusion approach (Concatenated input, No attn btw views), and safety controller (No safety-contrl). The results demonstrated the effectiveness of each module.

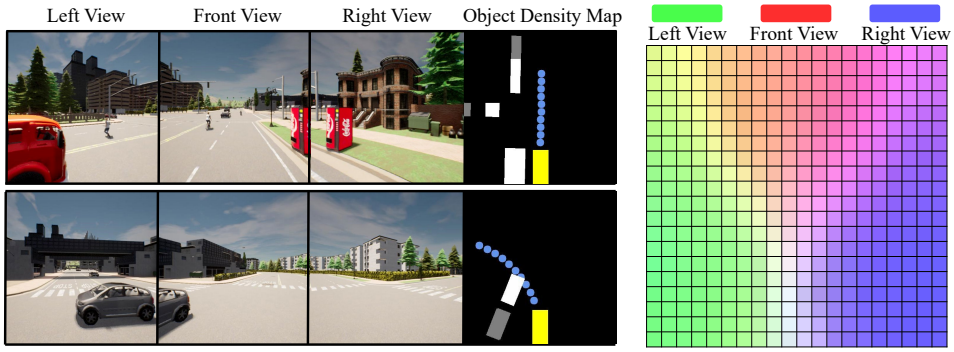
4.3 Ablation study

On the Town05 Long benchmark, we investigated the influence of different sensor inputs, sensor/position embedding, sensor fusion approach, and safety controller. In addition to three metrics, we also present infraction details for comprehensive analysis.

Sensor inputs As in Table 2, we evaluated the performance when different combinations of sensor inputs are utilized. In the benchmark, adversarial events such as emerging pedestrians or bikes from occluded regions are very common. Since the baseline F only uses the front RGB image input, it is hard to notice obstacles on the sides of the ego vehicle. Therefore, though the baseline F achieved a high route completion ratio, it significantly downgraded in the driving score and infraction score. When the left and right cameras are added, $F + LR$ achieved a higher driving score and infraction score with effectively reduced collision rate. Since traffic lights are located on the opposite side of the intersection, they are usually too far to be detectable in the original front image. Adding a focusing view for distant sight, $F + LR + Fc$ reduced the red light infraction rate by 80% compared to $F + LR$, resulting in a higher driving score and infraction score. Equipped with another Lidar input for additional 3D context, our Interfuser $F + LR + Fc + Li$ further reduced the collision rate and red light infractions, leading to the highest driving score and infraction score .

Sensor Embedding and Positional Encoding In InterFuser, we added the sensor embedding and the positional encoding to help the transformer distinguish tokens in different sensors and positions within one sensor. As in Table 3, when these operations are removed, the ablation "No sensr/pos embd" had the driving score and the infraction score dropped by 3% and 6% respectively.

Sensor Fusion Methods In Table 3, we also evaluated the performance when different sensor fusion methods are applied. There are two common methods to fuse multi-view or multi-modal inputs: 1) directly concatenating three RGB views along the channel dimension, as in LBC [5] and IA [6]. This ablation "Concatenated input" is assumed to insufficiently fuse information and turned out to have the driving score and infraction score lower by 58% and 63%. 2) multi-stage fusion of images and LiDAR by transformer as in TransFuser [8]. Though applying the attention mechanism, this method does not consider multi-view fusion and can have problems processing multi-view multi-modal inputs. We implemented this approach by designing a mask to disallow attention between different views in the transformer encoder, resulting in drops of 9% and 15% on the driving score and the infraction score.



(a) (b)

Figure 3: (a) Two cases of how our method predicts waypoints and recovers the traffic scene. Blue points denote predicted waypoints. The yellow rectangle represents the ego vehicle, and white/grey rectangles denote the current/future positions of detected objects. (b) Visualization of attention weights between object density map queries and features from different views.

Safety-enhanced controller As in Table 3, when the safety controller is removed and our model directly outputs waypoints, the driving score and the infraction score dropped significantly by 15%. and 22%. Besides, different driving preferences can be generated by assigning different safety factors in the safety controller. As in Fig. 4 of Appendix B, the agents behaves more conservatively with a higher safety factor, leading to higher driving score and infraction score, and lower route completion.

4.4 Visualization

In Fig. 3 (a), we visualized two cases to show how our method predicts the waypoints and recovers the surrounding scene. In Fig. 3 (b), we also visualize the attention weights on 20×20 object density map in the transformer decoder, to show how our method aggregates information from multiple views. As in Fig. 3 (b), our model extracts features from different views for queries at different grids of the object density map. For example, the queries on the left of the map mostly attend to the features from the left views. More case visualizations can be found in Fig. 5 and Fig. 6 in the Appendix.

5 Limitation

Though currently ranking the first on public CARLA Leaderboard, our method still has traffic infractions. Visualization of failure cases can be found in Fig. 7 in the appendix. Currently, we used two-threshold criteria for detection, and dynamic propagation with moving average for trajectory prediction. In the future, more advanced detection and prediction models can be used. Extending the current deterministic form to a probabilistic form can also help to deal with the uncertainty and multi-modality in detection and prediction. Besides, all this work happened in simulation, where driving scenarios are limited. On actual roads where countless driving situations exist, enhancing the generalizability of our method would be vital for scalable deployment.

6 Conclusion

We present InterFuser, a new design for autonomous driving based on an interpretable sensor fusion transformer. The driving agent obtains a perception of global context from multi-view multi-modal inputs and generates some interpretable intermediate outputs. Using the interpretable outputs and safety-enhanced controller, InterFuser sets the new performance upper-bound in complex and adversarial urban scenarios on the CARLA. Importing temporal information into the pipeline is our future work. With the historical frames, the model can get a more accurate and stable prediction for the position and future trajectory of the other objects. Given that our method is flexible and generic, it would be interesting to explore it further with different types of interpretable intermediate outputs, or change the controller implemented in another way.

References

[1] N. E. Boudette. Teslas self-driving system cleared in deadly crash. The New York Times, 2017.

[2] S. Gibbs. Ubers self-driving car saw the pedestrian but didnt swerve–report. The Guardian, 2018.

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

[4] M. Liang, B. Yang, S. Wang, and R. Urtasun. Deep continuous fusion for multi-sensor 3d object detection. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 641–656, 2018.

[5] D. Chen, B. Zhou, V. Koltun, and P. Krähenbühl. Learning by cheating. In *Conference on Robot Learning*, pages 66–75. PMLR, 2020.

[6] M. Toromanoff, E. Wirbel, and F. Moutarde. End-to-end model-free reinforcement learning for urban driving using implicit affordances. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7153–7162, 2020.

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

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

[9] Z. Li, M. Pan, T. Zhang, and X. Li. Testing dnn-based autonomous driving systems under critical environmental conditions. In M. Meila and T. Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 6471–6482. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/li21r.html.

[10] A. Filos, P. Tigkas, R. McAllister, N. Rhinehart, S. Levine, and Y. Gal. Can autonomous vehicles identify, recover from, and adapt to distribution shifts? In *International Conference on Machine Learning*, pages 3145–3153. PMLR, 2020.

[11] C. Liu, T. Arnon, C. Lazarus, C. Strong, C. Barrett, and M. J. Kochenderfer. Algorithms for verifying deep neural networks. *arXiv preprint arXiv:1903.06758*, 2019.

[12] A. Radulescu, Y. Niv, and I. Ballard. Holistic reinforcement learning: the role of structure and attention. *Trends in cognitive sciences*, 23(4):278–292, 2019.

[13] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun. Carla: An open urban driving simulator. In *Conference on robot learning*, pages 1–16. PMLR, 2017.

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

[15] A. Prakash, A. Behl, E. Ohn-Bar, K. Chitta, and A. Geiger. Exploring data aggregation in policy learning for vision-based urban autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11763–11773, 2020.

[16] A. Zhao, T. He, Y. Liang, H. Huang, G. V. d. Broeck, and S. Soatto. Sam: Squeeze-and-mimic networks for conditional visual driving policy learning. *arXiv preprint arXiv:1912.02973*, 2019.

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

[18] J. Chen, S. E. Li, and M. Tomizuka. Interpretable end-to-end urban autonomous driving with latent deep reinforcement learning. *IEEE Transactions on Intelligent Transportation Systems*, 2021.

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

[20] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[21] M. Chen, A. Radford, R. Child, J. Wu, H. Jun, D. Luan, and I. Sutskever. Generative pretraining from pixels. In *International Conference on Machine Learning*, pages 1691–1703. PMLR, 2020.

[22] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021.

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

[24] Y. Jiang, S. Chang, and Z. Wang. Transgan: Two transformers can make one strong gan. *arXiv preprint arXiv:2102.07074*, 1(2):7, 2021.

[25] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021.

[26] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030*, 2021.

[27] A. Bozic, P. Palafox, J. Thies, A. Dai, and M. Nießner. Transformerfusion: Monocular rgb scene reconstruction using transformers. *Advances in Neural Information Processing Systems*, 34, 2021.

[28] Y. Hu, W. Zhan, and M. Tomizuka. Probabilistic prediction of vehicle semantic intention and motion. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 307–313. IEEE, 2018.

[29] L. Wang, L. Sun, M. Tomizuka, and W. Zhan. Socially-compatible behavior design of autonomous vehicles with verification on real human data. *IEEE Robotics and Automation Letters*, 6(2):3421–3428, 2021.

[30] K. Leung, E. Schmerling, M. Zhang, M. Chen, J. Talbot, J. C. Gerdes, and M. Pavone. On infusing reachability-based safety assurance within planning frameworks for human–robot vehicle interactions. *The International Journal of Robotics Research*, 39(10-11):1326–1345, 2020.

[31] J. F. Fisac, A. Bajcsy, S. L. Herbert, D. Fridovich-Keil, S. Wang, C. J. Tomlin, and A. D. Dragan. Probabilistically safe robot planning with confidence-based human predictions. *arXiv preprint arXiv:1806.00109*, 2018.

[32] L. Wang, Y. Hu, L. Sun, W. Zhan, M. Tomizuka, and C. Liu. Hierarchical adaptable and transferable networks (hatn) for driving behavior prediction. *arXiv preprint arXiv:2111.00788*, 2021.

[33] P. Mirowski, R. Pascanu, F. Viola, H. Soyer, A. J. Ballard, A. Banino, M. Denil, R. Goroshin, L. Sifre, K. Kavukcuoglu, et al. Learning to navigate in complex environments. *arXiv preprint arXiv:1611.03673*, 2016.

[34] N. Rhinehart, R. McAllister, K. Kitani, and S. Levine. Precog: Prediction conditioned on goals in visual multi-agent settings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2821–2830, 2019.

[35] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig. Virtual worlds as proxy for multi-object tracking analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4340–4349, 2016.

[36] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[37] J. L. Ba, J. R. Kiros, and G. E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

[38] L. Wang, Y. Hu, and C. Liu. Online adaptation of neural network models by modified extended kalman filter for customizable and transferable driving behavior prediction. *arXiv preprint arXiv:2112.06129*, 2021.

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

[40] L. Wang, Y. Hu, L. Sun, W. Zhan, M. Tomizuka, and C. Liu. Transferable and adaptable driving behavior prediction. *arXiv preprint arXiv:2202.05140*, 2022.

[41] H. Zhao, J. Gao, T. Lan, C. Sun, B. Sapp, B. Varadarajan, Y. Shen, Y. Shen, Y. Chai, C. Schmid, et al. Tnt: Target-driven trajectory prediction. *arXiv preprint arXiv:2008.08294*, 2020.

[42] B. Pang, T. Zhao, X. Xie, and Y. N. Wu. Trajectory prediction with latent belief energy-based model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11814–11824, 2021.

[43] C. Liu and M. Tomizuka. Control in a safe set: Addressing safety in human-robot interactions. In *Dynamic Systems and Control Conference*, volume 46209, page V003T42A003. American Society of Mechanical Engineers, 2014.

[44] C. team. Carla autonomous driving leaderboard. https://leaderboard.carla.org/, 2020. Accessed: 2021-02-11.

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

[46] 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.

[47] R. Chekroun, M. Toromanoff, S. Hornauer, and F. Moutarde. Gri: General reinforced imitation and its application to vision-based autonomous driving. *arXiv preprint arXiv:2111.08575*, 2021.

[48] Expert drivers for autonomous driving. urlhttps://kait0.github.io/files/master_thesis_bernhard_jaeger.pdf, 2021.

[49] D. Chen, V. Koltun, and P. Krähenbühl. Learning to drive from a world on rails. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15590–15599, 2021.

[50] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[51] I. Loshchilov and F. Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2018.

[52] I. Loshchilov and F. Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.