2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Multi-Horizon Lane Change Maneuver Prediction Using Multi-Modal Transformers

Petrit Rama[1], Praveen Kumar Gummadi[2] and Naim Bajcinca[3]

Abstract— **Predicting lane change maneuvers is essential for ensuring safe autonomous driving, especially in complex urban environments. Building upon prior multi-modal and graph-based approaches, this work introduces a novel transformer-based architecture for multi-horizon lane change prediction that jointly estimates the lane change maneuver and the lane change phase. The proposed model integrates visual information from surround-view cameras, semantic masks for free space and lane markings, interaction-aware graph representations, and ego-vehicle state signals, within a unified transformer framework to capture spatial-temporal dependencies. In addition, a multi-level uncertainty estimation branch quantifies confidence at the level of modality, fusion, and prediction, to enhance interpretability and reliability. Experiments are conducted on WylonSet++, an extended in-house dataset collected using an instrumented test vehicle, annotated for lane change behavior analysis and maneuver phase transitions. The dataset comprises synchronized front-facing camera images, left and right surround-view camera images, together with vehicle state data. The dataset contains approximately 600 lane change sequences, providing the foundation for this study. Extensive evaluations demonstrate strong performance in anticipating lane change maneuvers and phase progression across short- and long-term prediction horizons in diverse real-world traffic scenarios.**

I. Introduction

Lane change maneuvers are inherently dynamic and context-sensitive, influenced by the ego-vehicle's goals as well as the behavior of surrounding traffic agents and road geometry. Modeling driving intent in complex urban scenarios is particularly challenging due to uncertainty in the behavior of nearby agents and the interactive nature of traffic. These spatial interactions create complex dependencies and are therefore important for accurate maneuver prediction. This multi-step reasoning process reflects the hierarchical structure of human driving, where strategic decision-making precedes planning. Achieving similar competence in autonomous driving (AD) requires models that integrate scene understanding and multi-agent interactions.

Graph-based and vectorized scene representations have gained prominence due to their flexibility in modeling interactions among multiple agents. Graph Neural Networks (GNNs) have shown strong performance in capturing features from such interactions for motion prediction tasks. GRIP [1], VectorNet [2], LaneGCN [3], and Trajectron++ [4] introduce graph structures where agents and map elements, together with their trajectories and interactions, are encoded as vectors, nodes, or edges of the graph, using GNNs or graph convolution for spatial reasoning and to capture traffic context. These approaches rely on structured HD maps or preprocessed semantic data to model the traffic scene.

Deep learning has become the dominant approach for motion prediction, especially trajectory prediction. DESIRE [5], Lee et al. [6], Wei et al. [7], MultiPath++ [8] introduce deep learning frameworks that combine convolutional neural networks (CNNs) to extract visual features, and recurrent neural networks (RNNs) to extract temporal features. These works primarily focus on trajectory prediction or lane change detection, in a bird's-eye view, and often for surrounding vehicles, rather than from the ego-vehicle perspective.

Transformers [9] have been successfully adapted for motion prediction in AD due to their ability to model long-range dependencies and capture global context through self-attention. Scene Transformer [10], MultiModal Transformer [11], HiVT [12], Hu et al. [13] adopt transformer architecture to model perception and planning, but also process past agent trajectories, map data, or inter-agent interactions. Most transformer-based approaches focus on trajectory prediction using agent positions and map features, while comparatively few works address ego-centric maneuver intent prediction with structured phase modeling.

Although motion prediction for AD has made significant progress, most research focuses on trajectory prediction. Consequently, benchmark datasets have been primarily focused on trajectory prediction [14]–[16]. Datasets such as BLVD [17] and HDD [18] provide limited annotations for maneuvers and contain only a small number of lane change instances. As a result, they lack the variability and complexity needed to study the driving behavior of lane changes in diverse traffic scenarios. To address this gap, we introduce WylonSet++, an extended and enriched version of the WylonSet dataset originally presented in [19]. Collected using an instrumented test vehicle, WylonSet++ covers a wide range of urban and rural traffic scenarios, offering a significantly broader distribution of lane change maneuvers. The dataset includes high-resolution front-facing, left and right area-view camera images, ego-vehicle state data, and lane markings. Each frame in the dataset is labeled with two sets of labels: lane change maneuver (`LCM`) and lane change phase (`LCP`). The `LCM` labels consist of three classes: lane keeping (`LK`), left lane change (`LLC`) and right lane change (`RLC`). The `LCP` labels follow a structured finite-state transition model that captures valid maneuver progression through preparation, execution, and exit stages. WylonSet++ serves as the foundation for this study, enabling the development

[1,2,3] {petrit.rama, praveen.gummadi, naim.bajcinca}@rptu.de
[1,2,3] Department of Mechanical and Process Engineering, Rheinland-Pfälzische Technische Universität Kaiserslautern-Landau, Gottlieb-Daimler-Straße 42, 67663 Kaiserslautern, Germany.

and evaluation of models that jointly predict lane change maneuvers and phase progression for decision-making.

Prior work done in [20]–[22] on maneuver prediction has explored the use of GNNs to capture inter-agent spatial interactions and RNNs for temporal reasoning, respectively. Building on these ideas and inspired by our recent work on lane change prediction [19], which emphasized the benefits of scene graphs and attention mechanisms, we recognize the need for more expressive temporal modeling and feature fusion strategies for holistic scene understanding and accurate lane change prediction. In particular, these earlier approaches typically relied on simple concatenation of multi-modal features, losing their semantic richness in long traffic sequences and limiting the model's ability to capture expressive dependencies between interacting traffic agents. To address this, transformers [9] are adopted as the core module, leveraging the multi-head attention mechanism to fuse adaptively multi-modal features, learn long-range dependencies, and maintain context-aware reasoning through ego-centric modeling. These aspects are crucial for anticipating lane change maneuvers and phase transitions in dynamic traffic environments throughout the observation time. We propose a transformer-based architecture that jointly predicts `LCM` and `LCP` across multiple future time horizons, enabling interpretable decision-making and structured multi-horizon intent anticipation rather than single-step classification. Our architecture fuses multiple input modalities, including front-facing and left/right surround camera images, semantic segmentation masks for free space and lane markings, ego-vehicle state signals, and spatial interaction graphs built from detected traffic entities using an object detection algorithm. The graph-based scene representation models the relative spatial relationships and object semantics of traffic entities in a flexible and scalable form, enabling the network to learn complex interactions among traffic agents. While the present study focuses on WylonSet++, the proposed architecture is modality-flexible and can operate with partial input configurations, as demonstrated in our modality ablation study.

The main contributions of this work are as follows:

- A transformer-based multi-modal architecture for structured multi-horizon lane change prediction that integrates visual features, graph-based interaction modeling, semantic scene context, and ego-vehicle state signals through attention-based modality fusion;
- A joint prediction framework for lane change maneuver class (`LCM`) and lane change phase (`LCP`), enabling progressive and interpretable intent anticipation;
- A multi-level uncertainty estimation branch (`UEB`) that quantifies confidence at modality, fusion, and prediction levels to support safety-aware decision-making;
- A comprehensive experimental evaluation including feature fusion ablations, modality ablations, horizon-wise metrics, and event-level temporal analysis for assessing anticipation quality and timing consistency;
- The introduction of *WylonSet++*, a multi-modal dataset with dense lane change maneuver and phase annotations collected across diverse real-world traffic scenarios.

II. Methodology

Our approach leverages multiple modalities, including camera streams that capture the traffic scene, semantic masks of free space and lane markings, interaction graphs that encode spatial relationships among traffic participants, and ego-vehicle state signals (e.g., steering angle, yaw rate). Each modality provides a complementary aspect of scene understanding, each processed using a dedicated encoder based on transformer architectures. This modular design improves representation capacity and interpretability for decision-making.

Assuming a straight driving objective from the planning module, the model predicts two complementary outputs for the ego-vehicle over multiple future time horizons. The first output is the Lane Change Maneuver (`LCM`), with three classes: lane keeping (`LK`), left lane change (`LLC`), and right lane change (`RLC`). The second output is the Lane Change Phase (`LCP`), which follows a structured finite-state transition model that captures the progression through the preparation, execution, and completion phases of `LCM`. Modeling `LCP` improves interpretability by representing how maneuvers unfold over time through structured and explainable transitions.

Although `LCP` is defined through a finite-state transition structure at the dataset level, we do not impose hard transition constraints during inference. Instead, consistency across phases is implicitly encouraged through sequential modeling, horizon-wise loss weighting, and temporal regularization. In practice, invalid transitions occur rarely due to structured supervision during training. Incorporating explicit transition masking or constrained decoding is left for future work.

A. Problem Formulation

Given a sequence of T past observations $\mathcal{X}_{t-T:t}$, the objective is to jointly predict the future `LCM` and `LCP` of the ego-vehicle across $H = 6$ discrete future time horizons. Each horizon is spaced by a fixed temporal interval $\Delta_h = 0.5\,\text{s}$, resulting in a total prediction window of $H \cdot \Delta_h = 3.0\,\text{s}$. Each input sequence $\mathcal{X}_{t-T:t}$ consists of temporally aligned multi-modal features: multi-camera views $\mathcal{I}_t$, semantic masks $\mathcal{M}_t$, interaction graphs $\mathcal{G}_t$, and ego-vehicle state signals $\mathcal{S}_t$. The prediction model $f(\cdot)$ maps the input sequence to a multi-horizon output for each future time step $h \in 1, \ldots, H$:

$$\left(\hat{\mathbf{y}}^{\text{LCM}}_{1:H},\ \hat{\mathbf{y}}^{\text{LCP}}_{1:H}\right) = f\left(\mathcal{X}_{t-\mathcal{T}:t}\right), \tag{1}$$

where $\hat{\mathbf{y}}^{\text{LCM}}_{1:H} \in \mathbb{R}^{H \times C_{\text{LCM}}}$ and $\hat{\mathbf{y}}^{\text{LCP}}_{1:H} \in \mathbb{R}^{H \times C_{\text{LCP}}}$ denote the predicted maneuver and phase probability distribution.

B. Model Architecture

An overview of our proposed transformer-based architecture is shown in Fig. 1. The model processes T past time steps of the following input modalities at time t:

- multi-camera images defined as $\mathcal{I}_t = \{\mathcal{I}^v_t\}$, where $v \in \{F, L, R\}$ denotes the camera view (front, left, right);
- semantic masks for each camera view, six masks in total, defined as $\mathcal{M}_t = \{\mathcal{M}^{vm}_t\}$, where $v \in \{F, L, R\}$ denotes the camera view (front, left, right) and $m \in \{D, L\}$ denotes the mask type (drivable free-space area or lane marking), extracted using HybridNets [23];

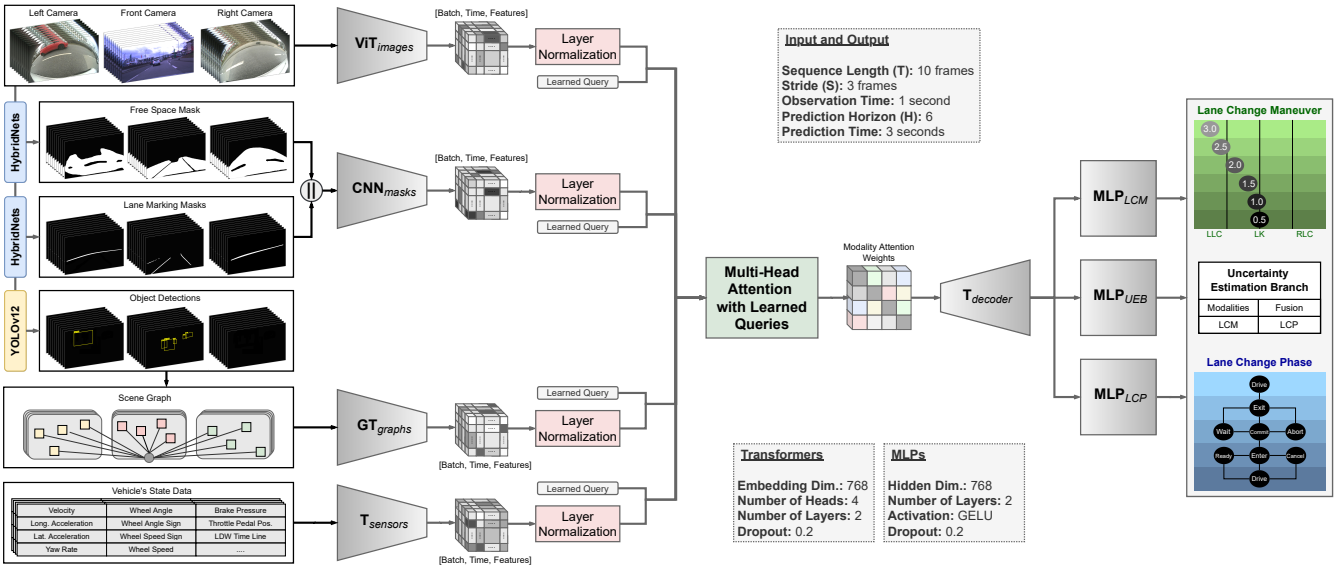

Fig. 1. Network architecture of the proposed transformer-based model for lane change maneuver and phase prediction.

- scene graphs $\mathcal{G}_t$, where nodes represent detected traffic entities with visual features from YOLOv12 [24], and edges encode pairwise spatial relationships;
- state data $\mathcal{S}_t$, comprising ego-vehicle's motion signals and lane-related features that encode road geometry.

Dedicated encoders are used to process each modality, tailored to its specific input domain. Each encoder serves as a feature extractor, producing temporally aligned embeddings for its input modality. Each transformer encoder learns temporal features and the sequential context, improving modularity and interpretability. The self-attention mechanism within each transformer enables selective focus over the whole observation window to capture important features. The resulting modality latent representations are fused via a multi-query temporal transformer decoder with learned per-modality query vectors, to produce multi-horizon predictions of LCM and LCP. The fusion process incorporates the uncertainty of UEB at the modality, fusion, and prediction levels to improve reliability and explainability of the model.

Image Encoder: The image encoder uses a ViT backbone to process input images from three camera views $v \in \{F, L, R\}$. A pre-trained $ViT\text{-}B/16$ model from TorchVision is applied to each frame within the sequence. Given the image sequence $\mathcal{I}^v_{t-T:t}$, visual embeddings are extracted as:

$$\mathbf{F}^v_{t-T:t} = \texttt{ViT}\left(\mathcal{I}^v_{t-T:t}\right), \tag{2}$$

where $\mathbf{F}^v_{t-T:t}$ represents the sequence of frame-wise visual embeddings extracted by the ViT backbone. The features from the three camera views are concatenated along the feature dimension and projected via a view-fusion MLP:

$$\mathbf{z}^{\text{image}}_{t-T:t} = MLP^{\text{image}}\left(\left[\mathbf{F}^F_{t-T:t}, \mathbf{F}^L_{t-T:t}, \mathbf{F}^R_{t-T:t}\right]\right). \tag{3}$$

The MLP uses $GELU$ activation and a dropout rate of 0.2, projecting the concatenated features into a shared embedding space of dimension $d = 768$. Additionally, a learnable scalar gate is assigned to each camera stream and optimized jointly with the network, for adaptive weighting of the camera view.

Mask Encoder: Semantic segmentation masks for drivable areas and lane markings are generated using HybridNets [23] from all three camera views. For each time step within the observation window, the binary masks are concatenated along the channel dimension and processed using a 3-layer CNN-based encoder. Let $\mathcal{M}_{t-T:t}$ mark the sequence of mask tensors. The corresponding embeddings are computed as:

$$\mathbf{z}^{\text{mask}}_{t-T:t} = \text{CNN}\left(\left[\mathcal{M}^{vm}_{t-T:t}\right]_{v\in\{F,L,R\},\, m\in\{D,L\}}\right). \tag{4}$$

The CNN encoder consists of convolutional layers with BatchNorm, ReLU, and 0.2 dropout, followed by max pooling and a linear projection to the shared embedding space.

Graph Encoder: Each traffic scene is represented as a spatial interaction graph capturing the visual context and surrounding entities relative to the ego-vehicle. Dynamic agents (e.g., vehicles, buses, cyclists) and static elements (e.g., traffic signs, traffic lights) are detected using YOLOv12 [24] from three synchronized camera views. Detected objects form graph nodes, merged into a unified ego-centric graph using the ego-vehicle as a common anchor node. Edges encode relative spatial proximity between entities in image space. For every time step t, a scene graph $\mathcal{G}_t$ is constructed, where the feature vector $f_i \in \mathbb{R}^{d_g}$ for node v_i is defined:

$$f_i = \left[x_i, y_i, w_i, h_i, s_i, e^c_i, e^v_i, v_i\right], \tag{5}$$

where (x_i, y_i) and (w_i, h_i) are bounding-box center coordinates and dimensions, s_i is the detection confidence score, $e^c_i \in \{0,1\}^C$ is a one-hot class encoding for C object categories (including the ego-vehicle class), $e^v_i \in \{0,1\}^3$ is a one-hot camera-view encoding, and $v_i \in \mathbb{R}^{d_v}$ denotes high-dimensional visual features obtained by average grouping the intermediate feature maps over the bounding box region of the YOLOv12 backbone. Edges are created between all pairs of nodes. Each edge is assigned a scalar weight e_{ij} computed from normalized Euclidean distances between object centers in image space, emphasizing close-range entities. Since edge weights are computed in normalized image space, cross-view

distances act as relative proximity heuristics rather than true geometric distances. Although this enables scalable multi-view fusion without BEV calibration, projecting graphs into BEV could provide more accurate spatial reasoning.

Each graph $\mathcal{G}_t$ is constructed per time step and is represented as a PyTorch Geometric graph. The node feature matrix $\mathbf{X}_t^{\mathcal{G}}$ is first projected into the shared embedding space of $d = 768$ using an MLP. Such graph representations are processed using a two-layer GT [25], with four attention heads and 0.2 dropout rate. GT extend transformers to graphs, where the attention weight is a function of graph features:

$$\hat{f}_i = MLP^{\text{graph}}(f_i), \quad \mathbf{z}_t^{\text{graph}} = \texttt{GT}(\hat{\mathcal{G}}_t). \tag{6}$$

GT captures spatial dependencies among entities. The graph-level feature is obtained by extracting the ego-node embedding, providing an ego-centric representation of the scene.

Sensor Encoder: The ego-vehicle state signals and lane-related data are obtained from the controller area network (CAN) bus. Let $\mathcal{S}_{t-T:t}$ denote the sequence of sensor measurements, which are normalized and one-hot encoded, then projected into the shared embedding space via an MLP. The projected sequence is processed by a 2-layer transformer encoder $\texttt{T}^{\text{sensor}}$ with four attention heads, hidden size $d = 768$, positional embeddings and layer normalization:

$$\mathbf{z}_{t-T:t}^{\text{sensor}} = \texttt{T}^{\text{sensor}}\left(MLP(\mathcal{S}_{t-T:t})\right). \tag{7}$$

Multi-Modal Feature Fusion: At each time step within the observation window, modality-specific embeddings are stacked and fused using a multi-head attention mechanism with learned modality-specific queries. Attention is applied independently at each time step across modality tokens. Let:

$$\mathbf{Z}_{t-T:t} = \left[\mathbf{z}_{t-T:t}^{\text{image}}, \mathbf{z}_{t-T:t}^{\text{mask}}, \mathbf{z}_{t-T:t}^{\text{graph}}, \mathbf{z}_{t-T:t}^{\text{sensor}}\right], \tag{8}$$

where $\mathbf{Z}_{t-T:t} \in \mathbb{R}^{T\times 4\times d}$ denotes the stacked modality embeddings. Fusion is performed via multi-head attention:

$$\mathbf{f}_{t-T:t} = \text{MultiHeadAttention}(\mathbf{Q}_{\text{mod}}, \mathbf{Z}_{t-T:t}, \mathbf{Z}_{t-T:t}), \tag{9}$$

where $\mathbf{Q}_{\text{mod}} \in \mathbb{R}^{4\times d}$ are learned modality-specific query vectors that allow adaptive weighting of input modalities.

Multi-Horizon Temporal Decoder: A learned set of horizon-specific future queries $\mathbf{Q}_{\text{horizon}} \in \mathbb{R}^{H\times d}$ are used as target in a transformer decoder $\texttt{T}^{dec}$, with the fused multi-modal temporal embeddings over the observation window $\mathbf{f}_{t-T:t} \in \mathbb{R}^{T\times d}$ used as memory, enabling cross-attention between past observations and future horizon embeddings:

$$\mathbf{h}_{1:H} = \texttt{T}^{dec}\left(\mathbf{Q}_{\text{horizon}}, \mathbf{f}_{t-T:t}\right). \tag{10}$$

The decoder consists of two layers with four attention heads and a dropout rate of 0.2, using $H = 6$ learned query embeddings for future LCM and LCP predictions. Separate prediction heads are applied to each horizon embedding:

$$\hat{\mathbf{y}}_{1:H}^{\text{LCM}} = MLP_{\text{LCM}}(\mathbf{h}_{1:H}) \in \mathbb{R}^{H\times C_{\text{LCM}}}, \tag{11}$$

$$\hat{\mathbf{y}}_{1:H}^{\text{LCP}} = MLP_{\text{LCP}}(\mathbf{h}_{1:H}) \in \mathbb{R}^{H\times C_{\text{LCP}}}. \tag{12}$$

Each prediction head is an MLP with $GELU$ activation, 0.2 dropout rate, and output size of $C_{\text{LCM}} = 3$ and $C_{\text{LCP}} = 8$.

Uncertainty Estimation Branch: To quantify prediction confidence and model reliability, we introduce a multi-level uncertainty estimation framework. The UEB consists of seven branches: four modality-level (image, mask, graph, sensor), one fusion-level, and two prediction-level (LCM, LCP). Each branch estimates a scalar confidence score, defined as:

$$\sigma_i = \text{Sigmoid}\left(MLP_i(\bar{\mathbf{z}}_i)\right) \in [0, 1], \tag{13}$$

where $\bar{\mathbf{z}}_i$ is the temporally aggregated latent features for branch i. Temporal aggregation uses mean pooling over the observation window or horizon embeddings for prediction-level branches. Each MLP_i consists of two linear layers with $GELU$ activation and 0.2 dropout. UEB branches are trained jointly with the main prediction task without explicit uncertainty supervision. Instead, uncertainty scores are learned implicitly from the classification loss. Thus, σ_i reflects the model's confidence related to feature consistency and prediction difficulty. Overall uncertainty is the unweighted mean of all seven branches, preventing dominance of any modality or prediction head. This design provides complementary signals about modality reliability and prediction stability, leveraged for downstream safety-aware decision-making.

III. Experimental Setup

A. Dataset: WylonSet++

Our experiments use the proprietary in-house dataset, denoted as *WylonSet++*, designed for lane change maneuver analysis. It extends the diversity and coverage of urban traffic scenarios, with increased traffic complexity, road variations, and construction zones, compared to our previous version [19]. *WylonSet++* was collected using an instrumented vehicle in Kaiserslautern, Germany, from October 2023 to February 2024 and from April to June 2025. Drivers were instructed to maintain normal driving behavior according to traffic rules, avoiding maneuvers other than lane changes.

The dataset comprises high-resolution front-facing camera images (30 Hz at 2048×864), left and right surround-view images (15 Hz at 1280×800). The CAN bus provides vehicle dynamics (e.g., braking, yaw rate, speed) and lane marking attributes (e.g., type, color). Camera streams and CAN data are synchronized using the front camera as the master clock.

Each frame is annotated with two prediction targets: LCM and LCP. Each instance of LCM is labeled as one of three maneuver classes: LK (64.65%), LLC (20.45%) or RLC (14.90%). This distribution reflects real-world driving behavior, where lane keeping dominates most of the driving time. In contrast, LCP is modeled as a structured finite-state transition model with eight discrete states, shown in Fig. 2. LCP is grouped into three phases: the preparation phase initiates readiness checks (Enter, Ready, Cancel), the lane change phase handles execution (Commit, Wait, Abort), and the exit phase finalizes the maneuver (Exit). States like Ready and Wait pause preparation or execution, while states Cancel and Abort represent interrupted maneuvers due to traffic.

LCM labels are generated automatically from turn indicators, activated at the start of lateral motion toward the target lane and deactivated once the vehicle stabilizes in the new

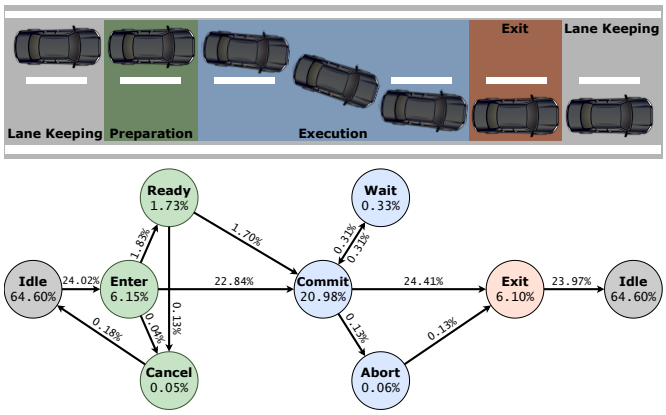

Fig. 2. LCM sequence and LCP finite state transition diagram.

lane. Subsequently, all LCM labels are verified manually. LCP events are labeled manually using time-to-maneuver, relative distance, and traffic context. Each lane change event is manually reviewed using a custom annotation tool to correct ambiguous transitions (e.g., delayed indicator activation or aborted maneuvers). To ensure label consistency, all sessions were independently reviewed by multiple annotators. The final *WylonSet++* dataset contains over 450 sessions, including about 360 labeled LLC events and 240 labeled RLC events.

B. Training Setup

Model training was conducted using AdamW (lr: 1×10^{-5}, weight decay: 1×10^{-4}, gradient clipping: 1.0). Experiments were conducted on a workstation with an Intel Xeon E5-2698 v4 CPU, 256GB RAM, and 4 × Tesla V100-DGXS-32GB.

Dataset Handling: Input sequences use a time window of $T = 10$ past observations. Frames are sampled with a stride density $S = 3$, skipping frames for more efficient processing. The stride acts as a temporal downsampling method, yielding a 10 Hz sampling rate over a past observation window of $W = 1.0$ s. For starting index i, sequences are constructed as $[i, i+S, i+2S, ..., i+(T-1)S]$. Sequences use jump parameter $J = 4$ between starting indices to reduce overlap. Each sequence ends at reference time t_0, from which the model predicts LCM and LCP at $H = 6$ future time horizons spaced 0.5 s apart (3.0 s total). A study of temporal parameters T, S, and W is presented in IV-C. A selective downsampling strategy is applied to mitigate the class imbalance problem of LK dominance and balance the dataset. A sequence is categorized as LK if all T past frames of the sequence and H future horizons correspond to lane keeping with no active lane change phase. A proportion $\rho \in [0, 1)$ of LK sequences is randomly discarded, with $\rho = 0.8$ in our experiments. After downsampling, the dataset is split by driving sessions into 70% training set, 15% validation set, and 15% test set.

Loss Function: The model is trained using a weighted cross-entropy loss $\mathcal{L}_{\text{CE}}$ that jointly optimizes LCM and LCP predictions across $H = 6$ discrete future time horizons. At each horizon $h \in \{1, \ldots, H\}$, the model outputs class probability distributions $\hat{\mathbf{y}}_h^{\text{LCM}}$ and $\hat{\mathbf{y}}_h^{\text{LCP}}$. To balance short- and medium-term predictions, an exponentially decaying weighting scheme is applied across horizons. Temporal weights w_h follow normalized exponential decay $\gamma = 0.5$, emphasizing earlier horizons. Task-specific weights $\alpha_{\text{LCM}} = 0.66$ and $\alpha_{\text{LCP}} = 0.34$ are applied to balance the relative importance of maneuver and phase predictions. The main loss is defined as:

$$\mathcal{L}_{\text{main}} = \sum_{h=1}^{H} w_h \left(\alpha_{\text{LCM}} \ell_h^{\text{LCM}} + \alpha_{\text{LCP}} \ell_h^{\text{LCP}}\right), \tag{14}$$

$$\ell_h^{\text{LCM}} = \mathcal{L}_{\text{CE}}(\hat{\mathbf{y}}_h^{\text{LCM}}, \mathbf{y}_h^{\text{LCM}}), \quad \ell_h^{\text{LCP}} = \mathcal{L}_{\text{CE}}(\hat{\mathbf{y}}_h^{\text{LCP}}, \mathbf{y}_h^{\text{LCP}}). \tag{15}$$

To improve training stability and interpretability, we adopt three attention regularization terms from prior work. *Entropy regularization* L_{ent} is used to prevent attention collapse, encouraging sparsity and more uniform entropy attention across modalities [26]. *Diversity regularization* L_{div}, a form of cosine-similarity regularizer for attention queries, is adopted to make different queries or heads attend to different modalities, encouraging complementary attention patterns across modalities [26]. *Temporal consistency regularization* L_{temp} is used to enforce smooth horizon-wise predictions over consecutive time steps [27]. The overall regularization term:

$$\mathcal{L}_{\text{attention}} = \lambda_{\text{ent}} L_{\text{ent}} + \lambda_{\text{div}} L_{\text{div}} + \lambda_{\text{temp}} L_{\text{temp}}, \tag{16}$$

with weights $\lambda_{\text{ent}} = 0.05$, $\lambda_{\text{div}} = 0.03$, and $\lambda_{\text{temp}} = 0.1$. The final training loss is defined as the sum $\mathcal{L} = \mathcal{L}_{\text{main}} + \mathcal{L}_{\text{attention}}$.

Evaluation Metrics: We report standard classification metrics: accuracy, macro F1-score, weighted F1-score, and Matthews Correlation Coefficient (MCC). Metrics are computed for each class and prediction horizon. Horizon-wise evaluation provides insight into short-term responsiveness and long-term anticipation capability, enabling analysis of how predictive performance evolves across prediction horizons. To further assess temporal alignment and anticipation behavior, we evaluate event-wise metrics introduced in [28]. These metrics quantify temporal quality and safety implications of maneuver predictions at the event level. We report *Time-to-Maneuver*, *Delay*, *Overlap*, *Prediction Frequency*, and *Miss Rate*. Detailed definitions are provided in IV-C.

IV. Results and Discussion

A. Comparison of Attention-based Fusion Methods

We evaluate several attention-based feature fusion strategies for combining the four input modalities. Results are reported for the LCM and LCP predictions across all horizons.

Attention with Mean Pooling: Uses the mean of all modality features as a query to assign attention weights during aggregation, focusing on the most relevant modalities.

Attention with Learned Query: Employs a single learned query vector that attends to all modalities, optimized to extract a weighted context-aware combination of embeddings.

Attention with Gated Pooling: Uses sigmoid-activated neural networks to generate soft gate values for each modality, enabling dynamic and input-specific fusion.

Attention with Per-Modality Queries: Employs separate learned queries per modality, which are aggregated via mean pooling for diverse and modality-specific information.

Attention with Weighted Pooling: Learns adaptive attention weights for each modality to compute a weighted sum of features, prioritizing more relevant modalities per instance.

Table I summarizes the performance comparison. Mean pooling and a single learned query already achieve competitive performance, indicating that modality-aware attention is beneficial. Gated pooling performs slightly worse, suggesting that independent gating without cross-modality interaction may limit representational flexibility. The per-modality query strategy achieves the best overall performance (Acc: 97.15%, W-F1: 97.78%), outperforming other fusion approaches across all reported metrics. This result indicates that allowing each modality to attend independently encourages complementary feature extraction and richer cross-modal interactions. Improvements are consistent across macro and weighted F1, indicating robustness under class imbalance.

TABLE I

PERFORMANCE COMPARISON OF FEATURE FUSION STRATEGIES (IN %).

Fusion Strategy	Acc	W-F1	M-F1	MCC
Attention: Mean Pooling	95.37	96.68	96.33	94.83
Attention: Learned Query	94.89	96.45	96.07	94.55
Attention: Gated Pooling	94.40	96.01	95.64	93.70
Attention: Weighted Pooling	95.11	96.98	96.63	95.32
Attention: Query per Modality	**97.15**	**97.78**	**97.24**	**95.95**

B. Modality Ablation Study

An extensive modality ablation study evaluates the contribution of each modality: camera images $\mathcal{I}$, semantic masks $\mathcal{M}$, interaction graphs $\mathcal{G}$, and ego-vehicle state signals $\mathcal{S}$. All configurations use the per-modality query vectors as the attention fusion strategy, described in IV-A.

As shown in Table II, the four-modality configuration achieves the best performance, demonstrating the complementary benefits of integrating all modalities. Configurations including camera images consistently maintain strong performance ($\mathcal{I}\cdot\mathcal{M}\cdot\mathcal{G}$: 94.48%, $\mathcal{I}\cdot\mathcal{M}\cdot\mathcal{S}$: 94.16%, $\mathcal{I}\cdot\mathcal{G}\cdot\mathcal{S}$: 94.28%), while the configuration without images ($\mathcal{M}\cdot\mathcal{G}\cdot\mathcal{S}$: 69.19%) shows substantial degradation. This confirms that raw visual context is the dominant modality for robust maneuver prediction, while other modalities serve as complementary information sources. Single-modality settings further confirm this trend, with the image-only configuration (92.95%) again showing comparatively strong performance. In contrast, mask-only (61.82%), graph-only (45.28%), and sensor-only (66.91%) settings lack sufficient discriminative power.

Modality-level uncertainty from `UEB` reflects model confidence across configurations. Visual modality $\mathcal{I}$ exhibits relatively stable uncertainty (48–54%), suggesting a consistent contribution when present. Mask-based configurations show increased uncertainty in the absence of images ($\mathcal{M}\cdot\mathcal{G}\cdot\mathcal{S}$: 57.38%), reflecting reduced contextual reliability when visual context is missing. The graph-only configuration exhibits elevated uncertainty (52.19%), while the sensor uncertainty remains comparatively stable (48–53%). The uncertainty patterns reveal that the model learns to calibrate its modality-specific uncertainty based on the available information sources, with higher uncertainty (lower confidence) in modalities when they serve as primary sources and more conservative estimates when multiple modalities are available. Uncertainty values should be interpreted comparatively rather than absolutely, as the `UEB` outputs represent learned confidence indicators rather than calibrated probabilities.

Fusion-level uncertainty remains stable across configurations (48–53%), indicating robust multi-modal integration. Prediction-level uncertainty for `LCM` (46–56%) and `LCP` (48–55%) varies more noticeably, with higher uncertainty in single-modality setups ($\mathcal{G}$: 55.74% and 47.61%, $\mathcal{S}$: 52.08% and 54.52%). This behavior suggests that the model increases internal uncertainty when predictive evidence is limited. Overall uncertainty remains consistent (49–52%), computed as the unweighted mean across seven branches: four modality-level branches (image, mask, graph, sensor), one fusion-level branch, and two prediction-level branches (`LCM`, `LCP`), avoiding bias toward single modalities or tasks.

C. Temporal Window Study

This study analyzes the impact of different observation windows $W \in \{1\,s, 2\,s, 3\,s\}$ and temporal configurations ($T \times S$) on frame-wise classification and event-wise metrics for `LCM` prediction, described in detail below.

- Time-to-Maneuver (TTM) quantifies how early the model can anticipate lane changes. *Coverage* indicates the percentage of events successfully detected, while *Early Rates* indicate prediction timing precision relative to ground truth.
- Delay quantifies the temporal offset between predicted and actual lane changes. *Perfect Alignment* indicates predictions that occur at maneuver start. Positive delay indicates late predictions, and negative delay indicates early predictions.
- Overlap evaluates the temporal consistency between predicted and ground-truth maneuver intervals. *High Overlap* indicates predictions matching maneuver duration. *Consistency* measures the stability of overlap across prediction horizons.
- Frequency measures how often maneuvers are predicted per ground-truth event. For `LK`, this reflects false lane change predictions. *Stability* measures how consistently predictions occur. *Spurious Rate* indicates false positive prediction rates.
- Miss Rate, as a critical safety metric, measures the percentage of actual lane change events that were completely missed by the model, including missed detections and interventions.

Table III shows that within each observation window W, configurations with lower stride S (higher temporal resolution) generally achieve stronger performance. Configurations $[30 \times 1]$, $[30 \times 2]$, $[30 \times 3]$ consistently yield the highest weighted F1-scores, indicating that finer temporal sampling improves representation quality. Comparing across observation windows reveals limited gains beyond $W = 2\,s$. Yet, the comparison across different observation windows is challenging due to the varying sampling density S.

The TTM analysis shows consistently high coverage rates across configurations (96.7–99.3%). Early rates decrease systematically, from 19.63 - 22.99% in $W = 1\,\text{s}$ to 19.24 - 21.14% in $W = 2\,\text{s}$, and further to 15.43-16.72% in $W = 3\,\text{s}$. This trend highlights that longer temporal context and lower stride S values improve the timing of maneuver predictions.

Delay metrics demonstrate strong temporal alignment, with perfect alignment rates above 96% and mean delays consistently close to zero across all configurations.

TABLE II

COMPREHENSIVE MODALITY ABLATION STUDY WITH PERFORMANCE AND UNCERTAINTY METRICS (ALL IN %).

Modality	Metrics		Modality Uncertainty				Uncertainty Metrics			
	Acc	W-F1	Images	Mask	Graph	Sensor	Fusion	LCM	LCP	Overall
$\mathcal{I} \cdot \mathcal{M} \cdot \mathcal{G} \cdot \mathcal{S}$	**97.15**	**97.78**	**50.45±4.07**	**43.52±0.59**	**47.12±0.01**	**50.99±0.46**	**50.99±1.60**	**47.44±2.72**	**53.01±2.90**	**49.07±0.91**
$\mathcal{I} \cdot \mathcal{M} \cdot \mathcal{G}$	94.48	97.03	48.46±5.36	48.55±0.52	49.53±0.01	-	49.84±0.68	46.76±2.40	48.76±3.76	48.65±1.31
$\mathcal{I} \cdot \mathcal{M} \cdot \mathcal{S}$	94.16	96.41	53.50±3.32	44.61±0.72	-	48.20±0.40	48.28±0.99	49.52±3.32	49.85±2.31	48.99±0.99
$\mathcal{I} \cdot \mathcal{G} \cdot \mathcal{S}$	94.28	96.66	50.12±5.21	-	48.51±0.00	52.86±0.45	51.12±1.14	47.92±5.49	50.55±2.44	50.18±0.69
$\mathcal{M} \cdot \mathcal{G} \cdot \mathcal{S}$	69.19	79.16	-	57.38±2.04	51.32±0.01	49.40±0.72	49.20±0.76	50.36±2.97	50.35±3.91	51.33±1.10
$\mathcal{I}$	92.95	94.80	49.40±3.21	-	-	-	53.25±1.07	49.95±3.56	51.90±3.51	51.12±1.87
$\mathcal{M}$	61.82	71.20	-	50.72±2.66	-	-	51.59±1.15	45.50±2.79	49.19±5.55	49.25±1.96
$\mathcal{G}$	45.28	42.43	-	-	52.19±4.35	-	50.91±0.80	55.74±1.06	47.61±1.10	51.61±0.87
$\mathcal{S}$	66.91	75.99	-	-	-	51.35±0.90	50.21±0.93	52.08±3.08	54.52±4.44	52.04±1.63

TABLE III

EVENT-WISE EVALUATION METRICS FOR LANE CHANGE PREDICTION ACROSS DIFFERENT TEMPORAL WINDOW CONFIGURATIONS ($T \times S$).

	Observ. Window	1 second			2 seconds			3 seconds		
	$T \times S$	10×3	15×2	30×1	10×6	15×4	30×2	10×9	15×6	30×3
	Acc (%)	97.15	96.21	95.74	94.37	95.43	95.78	94.98	95.80	95.62
	W-F1 (%)	97.78	98.31	98.45	97.33	98.14	98.90	97.34	97.66	98.07
TTM	Coverage (%)	97.0994	97.6584	96.7168	98.0684	97.7544	99.2560	97.6672	98.6820	99.0640
TTM	Early Rate (%)	19.6302	22.9901	20.6506	19.2424	19.2955	21.1394	16.7197	15.8598	15.4331
TTM	LLC Mean (s)	1.0092±0.8	1.0297±0.8	1.0032±0.8	1.0170±0.8	0.9894±0.8	0.9907±0.8	0.9762±0.7	0.9254±0.7	0.9256±0.7
TTM	RLC Mean (s)	0.9963±0.8	1.0849±0.8	1.0322±0.8	0.9516±0.7	0.9802±0.8	1.0732±0.8	0.8537±0.7	0.9516±0.7	0.9203±0.7
Delay	Perfect Align. (%)	98.2930	97.8843	99.1513	96.6667	99.2343	99.1004	98.0922	98.4974	98.5827
Delay	LLC Mean (s)	-0.0046±0.06	0.0023±0.09	0.0011±0.02	0.0±0.08	0.0059±0.06	0.0±0.05	-0.0012±0.07	-0.0079±0.07	-0.0012±0.02
Delay	RLC Mean (s)	-0.0093±0.09	-0.0092±0.08	-0.0043±0.09	-0.0040±0.10	0.0022±0.03	-0.0042±0.05	-0.0109±0.07	0.0023±0.03	-0.0172±0.12
Overlap	High Overlap (%)	95.5801	96.0055	96.1696	95.5423	97.0060	98.3631	96.1120	97.3641	97.9719
Overlap	Consistency (%)	95.05±20.5	95.07±20.1	94.98±20.3	95.24±19.7	96.64±16.7	97.58±13.4	95.04±20.2	97.07±15.8	97.75±13.4
Overlap	Duration (s)	1.7072±0.9	1.7397±0.9	1.7544±0.9	1.7682±0.9	1.7874±0.9	1.7708±0.8	1.7970±0.9	1.8048±0.8	1.8448±0.8
Overlap	LLC Mean (%)	99.63±3.6	99.35±5.2	99.96±0.9	99.45±4.6	99.57±4.5	99.65±3.8	99.80±2.3	99.64±3.2	99.94±1.2
Overlap	RLC Mean (%)	99.56±4.2	99.57±5.1	99.39±5.0	99.28±5.3	99.78±3.3	99.79±3.2	99.74±2.8	99.85±2.3	98.71±7.2
Frequency	Stability (%)	99.55±11.8	99.70±9.9	99.33±7.8	99.25±12.1	99.90±12.1	99.65±6.9	100.14±16.5	99.70±11.8	99.77±11.3
Frequency	Spurious Rate (%)	0.06325	0.09517	0.06343	0.10040	0.03364	0.03377	0.16146	0.12681	0.01825
Frequency	LLC Mean (%)	100.22±23.7	98.63±23.7	96.10±21.9	102.54±22.8	99.22±20.5	100.25±12.7	102.78±24.9	103.47±19.8	95.66±24.9
Frequency	RLC Mean (%)	103.44±21.9	102.13±16.1	103.04±14.1	101.78±23.4	100.81±12.3	102.50±11.7	104.06±24.6	101.44±17.7	101.05±14.5
Frequency	LK Mean (%)	91.09±28.5	94.25±24.8	95.61±25.4	92.07±29.8	93.08±28.1	95.03±22.5	90.11±29.7	90.12±29.4	103.73±14.1
Miss	Missed Events	L:18, R:7	L:16, R:3	L:23, R:1	L:11, R:7	L:15, R:3	L:5, R:0	L:10, R:5	L:7, R:4	L:6, R:1
Miss	LLC Rate (%)	0.7246	0.6309	0.0089	0.0045	0.0062	0.0021	0.0043	0.0032	0.0026
Miss	RLC Rate (%)	0.2818	0.1183	0.0004	0.0029	0.0013	0	0.0022	0.0018	0.0004

Overlap analysis reveals high temporal consistency across configurations. The $[30 \times 2]$ configuration achieves the highest overlap rate (98.36%), suggesting that combining extended temporal context with relatively high sampling density yields the most stable estimates of maneuver duration.

Frequency metrics remain stable across configurations, with stability rates above 99% and very low spurious rates (0.02–0.16%), indicating robust control over false positives.

Miss rate analysis further highlights the influence of temporal resolution. The configuration $[30 \times 2]$ shows the lowest overall miss rate (0.0021%), with zero missed `RLC`. Lower stride S within a fixed observation window consistently reduces missed events. For example, within $W = 2\,\text{s}$, missed events decrease from 18 ($[10 \times 6]$) to 5 ($[30 \times 2]$), confirming improved reliability with denser temporal sampling.

D. Scenario Visualization

Fig. 3 illustrates qualitative inference results of the proposed model on two representative traffic scenarios from *WylonSet++*. The visualization combines multi-camera inputs, detected free space (green overlay) and lane markings (red line), main state signals from the ego-vehicle, predicted `LCM` and `LCP` probability distributions across future horizons. It illustrates modality importance analysis, modality attention weight matrix for cross-modal attention, and the quantification of prediction uncertainty for interpretability.

V. CONCLUSION

This work introduces a transformer-based architecture for multi-horizon lane change prediction, integrating multi-modal perception and interaction-aware reasoning. The model jointly predicts lane change maneuvers and phases using attention-based fusion with learned modality queries. A horizon-weighted training loss with attention regularization supports stable optimization and temporally consistent predictions. The multi-level uncertainty estimation of the model provides confidence indicators at modality, fusion, and prediction levels, improving interpretability for safety-critical decision-making. Extensive evaluation using classification metrics and event-wise temporal metrics demonstrates strong predictive performance and robust temporal alignment on the lane change dataset *WylonSet++*. Qualitative analysis

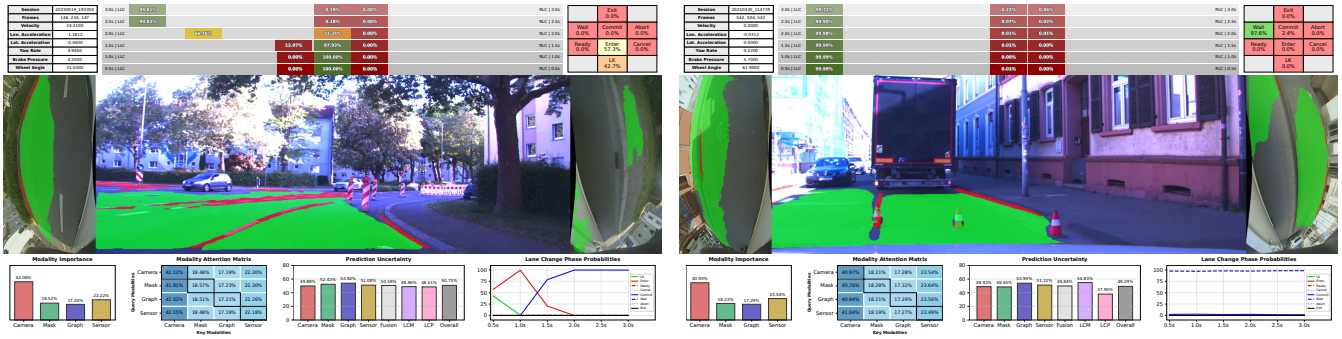

Fig. 3. (Left) Gradual LLC with LCP transition from LK to Commit. (Right) LLC prediction while maintaining the Ready phase due to opposing traffic.

shows the model's ability to separate maneuver intent from execution readiness while providing explainable predictions through attention visualization and uncertainty estimates. Although evaluation is conducted on *WylonSet++*, the strong performance of image-dominant configurations suggests potential transferability to publicly available datasets.

REFERENCES

[1] X. Li, X. Ying, and M. C. Chuah, "GRIP: Graph-based Interaction-aware Trajectory Prediction," in *2019 IEEE Intelligent Transportation Systems Conference, ITSC 2019, October 27-30, 2019*. IEEE, 2019.

[2] J. Gao, C. Sun, H. Zhao, Y. Shen, D. Anguelov, C. Li, and C. Schmid, "VectorNet: Encoding HD Maps and Agent Dynamics From Vectorized Representation," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, WA, USA, June 13-19, 2020*. Computer Vision Foundation / IEEE, 2020, pp. 11 522–11 530.

[3] M. Liang, B. Yang, R. Hu, Y. Chen, R. Liao, S. Feng, and R. Urtasun, "Learning Lane Graph Representations for Motion Forecasting," in *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part II*, ser. Lecture Notes in Computer Science, vol. 12347. Springer, 2020, pp. 541–556.

[4] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, "Trajectron++: Dynamically-Feasible Trajectory Forecasting with Heterogeneous Data," in *Computer Vision - ECCV 2020 - 16th European Conf., Glasgow, UK, August 23-28, 2020, Proceedings, Part XVIII*, ser. Lecture Notes in Computer Science, vol. 12363. Springer, 2020.

[5] N. Lee, W. Choi, P. Vernaza, C. B. Choy, P. H. S. Torr, and M. Chandraker, "DESIRE: Distant Future Prediction in Dynamic Scenes with Interacting Agents," in *2017 IEEE Conf. on Computer Vision and Pattern Recognition, CVPR 2017*. IEEE Computer Society, 2017.

[6] D. Lee, Y. P. Kwon, S. McMains, and J. K. Hedrick, "Convolution neural network-based lane change intention prediction of surrounding vehicles for ACC," in *20th IEEE Int. Conf. on Intelligent Transportation Systems, ITSC 2017, October 16-19, 2017*. IEEE, 2017.

[7] Z. Wei, C. Wang, P. Hao, and M. J. Barth, "Vision-Based Lane-Changing Behavior Detection Using Deep Residual Neural Network," in *2019 IEEE Intelligent Transportation Systems Conference, ITSC 2019, New Zealand, Oct. 27-30, 2019*. IEEE, 2019, pp. 3108–3113.

[8] B. Varadarajan, A. Hefny, A. Srivastava, K. S. Refaat, N. Nayakanti, A. Cornman, K. Chen, B. Douillard, C. Lam, D. Anguelov, and B. Sapp, "MultiPath++: Efficient Information Fusion and Trajectory Aggregation for Behavior Prediction," in *2022 Int. Conf. on Robotics and Automation, ICRA 2022, USA, May 23-27, 2022*. IEEE, 2022.

[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is All you Need," in *Advances in Neural Information Processing Systems 30, December 4-9, 2017, Long Beach, CA, USA*, 2017, pp. 5998–6008.

[10] J. Ngiam, V. Vasudevan, B. Caine, Z. Zhang, H. L. Chiang, J. Ling, R. Roelofs, A. Bewley, C. Liu, A. Venugopal, D. J. Weiss, B. Sapp, Z. Chen, and J. Shlens, "Scene transformer: A unified architecture for predicting future trajectories of multiple agents," in *The Tenth International Conference on Learning Representations, ICLR*, 2022.

[11] Y. Liu, J. Zhang, L. Fang, Q. Jiang, and B. Zhou, "Multimodal Motion Prediction With Stacked Transformers," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, June 19-25, 2021*. Computer Vision Foundation / IEEE, 2021, pp. 7577–7586.

[12] Z. Zhou, L. Ye, J. Wang, K. Wu, and K. Lu, "HiVT: Hierarchical vector transformer for multi-agent motion prediction," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[13] Y. Hu, J. Yang, L. Chen, K. Li, C. Sima, X. Zhu, S. Chai, S. Du, T. Lin, W. Wang, L. Lu, X. Jia, Q. Liu, J. Dai, Y. Qiao, and H. Li, "Planning-oriented Autonomous Driving," in *IEEE/CVF Conf. on Computer Vision and Pattern Recognition, CVPR 2023*. IEEE, 2023.

[14] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *Intl. Journal of Robotics Research, 32 (11)*, 2013.

[15] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The Cityscapes Dataset for Semantic Urban Scene Understanding," in *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, Las Vegas, NV, USA, June 27-30, 2016*. IEEE, 2016, pp. 3213–3223.

[16] X. Huang, X. Cheng, Q. Geng, B. Cao, D. Zhou, P. Wang, Y. Lin, and R. Yang, "The ApolloScape Dataset for Autonomous Driving," in *2018 IEEE Conf. on Computer Vision and Pattern Recognition Workshops, CVPR Workshops, USA, June 18-22, 2018*. IEEE, 2018, pp. 954–960.

[17] J. Xue, J. Fang, T. Li, B. Zhang, P. Zhang, Z. Ye, and J. Dou, "BLVD: Building A Large-scale 5D Semantics Benchmark for Autonomous Driving," in *Intl. Conf. on Robotics and Automation, ICRA, Montreal, QC, Canada, May 20-24, 2019*. IEEE, 2019, pp. 6685–6691.

[18] V. Ramanishka, Y. Chen, T. Misu, and K. Saenko, "Toward driving scene understanding: A dataset for learning driver behavior and causal reasoning," in *2018 IEEE Conf. on Computer Vision and Pattern Recognition, CVPR, June 18-22, 2018*. IEEE Computer Society, 2018.

[19] P. Rama and N. Bajçinca, "Multi-Modal Deep Learning Architecture Based on Edge-Featured Graph Attention Network for Lane Change Prediction," in *Proceedings of the 21st Int. Conf. on Informatics in Control, Automation and Robotics, ICINCO 2024, Porto, Portugal, Nov. 18-20, 2024, Volume 2*. SCITEPRESS, 2024, pp. 282–289.

[20] P. Rama and N. Bajcinca, "NIAR: Interaction-aware Maneuver Prediction using Graph Neural Networks and Recurrent Neural Networks for Autonomous Driving," in *Sixth IEEE Int. Conf. on Robotic Computing, IRC 2022, Naples, Italy, Dec. 5-7, 2022*. IEEE, 2022, pp. 368–375.

[21] P. Rama and N. Bajçinca, "Maneuver Prediction Using Traffic Scene Graphs via Graph Neural Networks and Recurrent Neural Networks," *Int. J. Semantic Comput.*, vol. 17, no. 3, pp. 349–370, 2023.

[22] P. Rama and N. Bajcinca, "MALE-A: Stimuli and Cause Prediction for Maneuver Planning via Graph Neural Networks in Autonomous Driving," in *26th IEEE Int. Conf. on Intelligent Transportation Systems, ITSC 2023, Spain, Sept. 24-28, 2023*. IEEE, 2023, pp. 3545–3550.

[23] D. Vu, B. Ngo, and H. Phan, "HybridNets: End-to-End Perception Network," *CoRR*, vol. abs/2203.09035, 2022.

[24] Y. Tian, Q. Ye, and D. Doermann, "YOLOv12: Attention-Centric Real-Time Object Detectors," *arXiv preprint arXiv:2502.12524*, 2025.

[25] V. P. Dwivedi and X. Bresson, "A Generalization of Transformer Networks to Graphs," *CoRR*, vol. abs/2012.09699, 2020.

[26] S. Zhai, T. Likhomanenko, E. Littwin, D. Busbridge, J. Ramapuram, Y. Zhang, J. Gu, and J. M. Susskind, "Stabilizing Transformer Training by Preventing Attention Entropy Collapse," in *Int. Conf. on Machine Learning, ICML 2023, July 2023, Honolulu, Hawaii, USA*, ser. Proceedings of Machine Learning Research, vol. 202. PMLR, 2023.

[27] S. Laine and T. Aila, "Temporal Ensembling for Semi-Supervised Learning," in *5th Int. Conf. on Learning Representations, ICLR 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[28] O. Scheel, N. S. Nagaraja, L. A. Schwarz, N. Navab, and F. Tombari, "Attention-based Lane Change Prediction," in *Int. Conf. on Robotics and Automation, ICRA 2019, May 20-24, 2019*. IEEE, 2019.