T^{2}SG: Traffic Topology Scene Graph for Topology Reasoning in Autonomous Driving

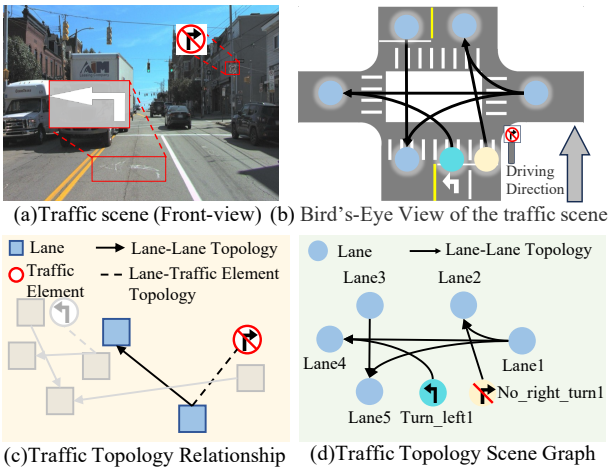
Changsheng Lv Mengshi Qi* Liang Liu Huadong Ma
State Key Laboratory of Networking and Switching Technology,
Beijing University of Posts and Telecommunications, China
{*lvchangsheng, qms, liangliu, mhd*}*@bupt.edu.cn*

Abstract

Understanding the traffic scenes and then generating high-definition (HD) maps present significant challenges in autonomous driving. In this paper, we defined a novel Traffic Topology Scene Graph (T^{2}SG), a unified scene graph explicitly modeling the lane, controlled and guided by different road signals (e.g., right turn), and topology relationships among them, which is always ignored by previous high-definition (HD) mapping methods. For the generation of T^{2}SG, we propose TopoFormer, a novel one-stage Topology Scene Graph TransFormer with two newly-designed layers. Specifically, TopoFormer incorporates a Lane Aggregation Layer (LAL) that leverages the geometric distance among the centerline of lanes to guide the aggregation of global information. Furthermore, we proposed a Counterfactual Intervention Layer (CIL) to model the reasonable road structure (e.g., intersection, straight) among lanes under counterfactual intervention. Then the generated T^{2}SG can provide a more accurate and explainable description of the topological structure in traffic scenes. Experimental results demonstrate that TopoFormer outperforms existing methods on the T^{2}SG generation task, and the generated T^{2}SG significantly enhances traffic topology reasoning in downstream tasks, achieving a state-of-the-art performance of 46.3 OLS on the OpenLane-V2 benchmark. Our source code is available at https://github.com/MICLAB-BUPT/T2SG.

1. Introduction

Understanding the traffic scene is the key component of autonomous driving. Except for detecting and recognizing individual elements, the vehicles need to infer the topology relationship among them. Conventional traffic scene understanding tasks, such as lane perception [30], road signal elements detection [4], and high definition (HD) mapping [15] focus primarily on the *isolated* elements (*i.e.*, map elements, and road signal elements) but miss the relationship among them. To model the unified HD mapping for lane perception and road signal elements association, the traffic topology reasoning task on OpenLane-V2 [33] has recently been proposed. As a map-like reasoning results shown in Figure 1(c), the reasoning task aims to create a topology graph of the detected elements and thus facilitate decision-making in the downstream tasks, such as ego planning [5] and motion prediction [24, 35, 45].



(a)Traffic scene (Front-view) (b) Bird's-Eye View of the traffic scene
(c)Traffic Topology Relationship (d)Traffic Topology Scene Graph

Figure 1. An example of a traffic scene understanding is illustrated as follows: (a) The traffic scene. (b) A BEV of the traffic scene, (c) The topology relationship proposed in TopoNet [13], (d) The T^{2}SG proposed in our work. Different from TopoNet, T^{2}SG can simultaneously model the whole relationships in the scene graph.

A primary challenge in traffic topology reasoning involves accurately modeling intricate traffic scene structures from multi-view camera inputs. Existing HD mapping methods [19] explicitly model the spatial relation between lanes but overlook the control and guide relationship between road signal elements and lanes. TopoNet [13] has

*Corresponding author: qms@bupt.edu.cn.

noticed this issue and proposed a graph-based method that treats the lanes and road signal elements as nodes and constructed a heterogeneous typology graph to describe the aforementioned relationships. However, this method neglects the control and guide information inherent in the traffic rules represented by each road signal. As illustrated in Fig 1 (b), a lane guided by the "Turn left" signal will only establish a connection with the lane left. Similarly, the "No right turn" signal also carries corresponding semantics.

To fully leverage this control and guide information to enhance lane centerline detection and traffic topology reasoning. Inspired by Scene Graph Generation [38], we explicitly model lanes guided by different road signals and their topology relationship using a unified traffic scene graph. In contrast to such scene graphs [38], the major challenge of the unified traffic scene graph we proposed is how to reason about relationships by simultaneously considering the spatial information of lanes and corresponding control and guide information of road signals. Compared to the independent traffic-lane and lane-lane topology reasoning tasks defined in OpenLane-V2 [33], the unified traffic scene graph facilitates joint learning of these two tasks.

Based on the above analysis, as illustrated in Fig 1 (d), we define a novel *Traffic Topology Scene Graph* (T^2SG), whose goal is to generate a visually-grounded scene graph from the input multi-view images. In the T^2SG, an object instance is characterized by a centerline with a corresponding category label and a relationship is characterized by a directed edge between two centerlines with a binary value $\{0,1\}$ indicating whether they are connected. To solve the complex relation reasoning problem, we propose a one-stage traffic topology scene graph generator **TopoFormer**, which stands for *Topology scene graph TransFormer*. TopoFormer contains a Lane Aggregation Layer that aggregates features according to the spatial proximity of the lane via geometry-guided self-attention. In this way, TopoFormer effectively obtains lane embeddings with the global context.

Furthermore, we capture the reasonable road structure among lanes in the traffic scene via the Counterfactual Intervention Layer (CIL), encompassing simple structures such as straight roads and more complex structures like crossroads and multi-way intersections. Previous methods [7, 19] focus on utilizing the spatial positions of centerlines to make *local* predictions of the lane relationships. However, these methods ignore the reasonable road structure in the real traffic scene, whereas joint reasoning with road structure can often resolve ambiguous relationships that arise from *local* predictions in isolation. Specifically, we consider the self-attention weights among lanes to signify the road structure and compare the factual structure (*i.e.*, the learned attention weights) with the counterfactual structure (*i.e.*, zero attention weights) on the ultimate prediction (*i.e.*, the output score). The proposed CIL can enhance the learned road structure's total indirect effect (TIE) on the prediction results.

Our main contributions can be summarized as follows:

(1) We propose the first unified Traffic Topology Scene Graph (T^2SG) to explicitly model the lanes, which are controlled and guided by different road signals, and the topology relationships among these lanes.

(2) We propose a Topology Scene Graph Transformer (TopoFormer) for T^2SG task, which captures global dependencies among lanes with a Lane Aggeration Layer.

(3) We introduce a Counterfactual Intervention Layer to emphasize the reasonable road structure influencing lane connectivity and effectiveness in topology reasoning tasks.

(4) We evaluate our TopoFormer in scene graph generation task and show it outperforms all state-of-the-art methods. Furthermore, we attain a 46.3 OLS on the traffic topology reasoning benchmark, OpenLane-V2 [33], demonstrating the effectiveness of the proposed T^2SG and TopoFormer for downstream tasks.

2. Related Work

Scene Graph Generation. Scene graphs were first introduced for Image Retrieval [11, 26], which broke down an image into its constituent objects, their attributes, and the relationships between them. The Visual Genome dataset [12] advanced image understanding, enabling scene graph extraction methods [6, 10, 23, 25]. Scene graphs are increasingly applied in tasks like image captioning [39] and visual question answering [34]. [32] introduced 3D scene graphs for object relationship perception, with subsequent research exploring GCN and Transformer-based methods [21] for point cloud registration [29] and scene generation [41]. Unlike prior approaches, T^2SG pioneers scene graphs in traffic scene understanding, modeling lanes as nodes and their connections as edges.

Traffic Topology Reasoning. STSU [2] introduced a lane topology reasoning approach for road structure comprehension, focusing on Bird's Eye View (BEV) lane centerline detection through three stages: BEV feature construction, centerline detection, and connection prediction. TPLR [3] and LaneGAP [16] refined centerline representation, enhancing continuity and shape accuracy. Online map construction methods like MapTR [15], VectorMapNet [19], BeMapNet [27], and Gemap [43] integrated lane topology reasoning with geometric information, modeling map elements such as lane lines, pedestrian crossings, and curbs. TopoNet addressed complex road scenarios by using Graph Neural Networks to connect driving lanes and traffic signs. TopoSeq [40] proposed randomized prompt-to-sequence learning for joint extraction of lane topology from Directed Acyclic Graphs and geometric lane graphs. TopoMLP [37] and TopoLogits [7] enhanced topology inference by leveraging lane spatial positions. In contrast,

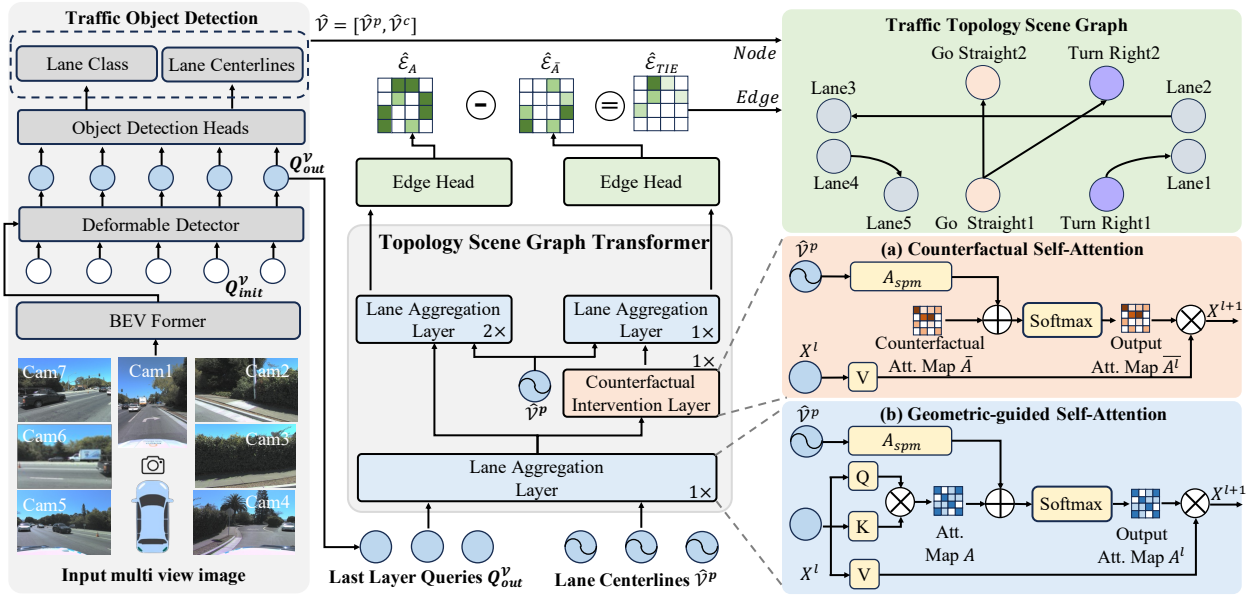


Figure 2. **The overview of our proposed TopoFormer.** Given the input multi-view images, we employ a DETR-like detector to identify lane objects with corresponding class and centerline coordinates. Subsequently, TopoFormer infers the relationships among these objects, which, along with the objects themselves, constitute the T^{2}SG. The main components of TopoFormer include two newly designed layers: (a) the Counterfactual Intervention Layer incorporating Counterfactual Self-Attention, and (b) the Lane Aggregation Layer incorporating Geometric-guided Self-Attention. Ultimately, the output of TopoFormer is a traffic topology scene graph, encapsulating the topological relationships among lanes, and guided by various road signals associated with the lanes.

our method employs a geometric-guided Lane Aggregation Layer, introducing spatial information to better illustrate lane relationships rather than augmenting lane features.

Counterfactual Intervention. Counterfactual Intervention (CI) is widely used in reasoning tasks like VQA [22], Re-ID [28], and scene understanding [9]. [22] modeled the physical knowledge relationships among different objects and apply them as counterfactual interventions to boost causal reasoning, while [28] enhanced Re-ID by applying CI to feature maps, maximizing task-relevant attention. Inspired by these, we leverage lane geometry to construct road structures and apply CI, maximizing the Total Indirect Effect (TIE) to encourage the model to learn more reasonable road structures for topology reasoning.

3. Proposed Approach

3.1. Overview

Problem Definition. We define a traffic topology scene graph, denoted as T^{2}SG, as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. This graph represents the traffic scene, detailing each lane's centerline coordinates, category, and connectivity with other lanes. Each lane $v_i \in \mathcal{V}$ has a lane category label v_i^c from a set of lane categories $\mathcal{C}_{lc}$, and an ordered list of 3D points v_i^p, which are sampled from the centerline along the direction of the lane at a fixed frequency (*i.e.*, 2Hz) following [33], denoted as $v_i = [v_i^c, v_i^p]$. Specifically, for a centerline $v_i^p = [p_1, p_2, ..., p_l]$, $p_1 = (x_1, y_1, z_1)$ represents the starting point of the lane, while $p_l = (x_l, y_l, z_l)$ denotes the ending point. The direction of a centerline, from the starting point to the ending point, denotes that a vehicle in this lane should follow the direction. Each edge $e_{ij} \in \mathcal{E}$ represents the connectivity relationship between lane v_i and lane v_j, where $i \neq j$ ensures that a lane does not connect to itself. Since a lane centerline is directed and represented as a list of points, the connection of two lanes means that the ending point of a centerline p_1 is connected to the starting point of another centerline p_l.

As illustrated in Figure 2, based on a DETR-liked [4] lane centerlines detector, the overall framework of our proposed method follows typical Transformer architecture but consists of two carefully designed components: *Lane Aggregation Layer* (**LAL**) captures global dependencies among lanes with geometry-guided self-attention; *Counterfactual Intervention Layer* (**CIL**) capture the reasonable road structure through counterfactual self-attention. For the edge $\mathcal{E}$, we use the total indirect effect $\hat{\mathcal{E}}_{\text{TIE}}$ calculated during the training stage as the output, while only using the normal predictions $\hat{\mathcal{E}}_A$ (A represents utilizing attention weights without counterfactual intervention.) in the inference stage. This will be detailed in the Training and Inference section.

Lane Centerline Detection. Given multi-view images

$\mathcal{I} = \left\{ I_i \in \mathbb{R}^{3 \times H_I \times W_I} \mid i = 1, 2, ..., N_I \right\}$ from N_I multi-camera views, where H_I and W_I represent the height and width of the input images, respectively. The backbone networks such as ResNet-50 [8] and FPN [17] are utilized to extract multi-view 2D features $\mathcal{F}_{2D}$. Based on 2D features $\mathcal{F}_{2D}$, we employ a simplified view Transformer, as proposed in BEVformer [14], to generate grid BEV feature $\mathcal{F}_{BEV}$, and use it as the input for the Deformable Detector [46], represented as:

$$\begin{aligned} Q_{out}^{\mathcal{V}} &= \text{DeformDETR}(Q_{init}^{\mathcal{V}}, \mathcal{F}_{BEV}), \\ \hat{\mathcal{V}}^p, \hat{\mathcal{V}}^c &= \text{Lane Head}(Q_{out}), \end{aligned} \tag{1}$$

where $Q_{init}^{\mathcal{V}}, Q_{out}^{\mathcal{V}} \in \mathbb{R}^{N \times 256}$ denote the initialized query and the output query from the final layer, respectively, and N signifies the number of queries. $\hat{\mathcal{V}} = [\hat{\mathcal{V}}^p, \hat{\mathcal{V}}^c]$, where $\hat{\mathcal{V}}^p = \{\hat{v}_i^p\}_{i=1}^N$ and $\hat{\mathcal{V}}^c = \{\hat{v}_i^c\}_{i=1}^N$ denote the predicted nodes in the graph $\hat{\mathcal{G}}$, and $\left\{ \hat{v}_i^p \in \mathbb{R}^{l \times 3}, \hat{v}_i^c \in C_{lc} \mid i = 1, 2, .., N \right\}$ signifies the ordered points list of the lane centerlines and class of the lane, respectively. The Lane Head$(\cdot)$ is constructed by two independent multilayer perceptron (MLPs) to predict the points of centerlines and the classification scores of lanes.

3.2. Lane Aggregation Layer

The Lane Aggregation Layer (LAL) is designed to leverage the geometric distance among the centerline of lanes to guide the aggregation of global structural information, which is a Transformer-encoder-like layer with the core component of Geometry-guided Self-Attention (GSA). Following [13], we utilize the output query $Q_{out}^{\mathcal{V}}$ as the input lane feature and predicted centerlines $\hat{\mathcal{V}}^p$ as the geometric information for the lane centerline. These are then fed into the Lane Aggregation Layer. The input lane features are passed through linear projections to be embedded into a d-dimensional hidden feature $X^0 \in \mathbb{R}^{N \times d}$. The output X^l is the feature encoded by l layers of LAL.

Geometry-guided Self-Attention (GSA) is proposed in the layer for the message passing in the graph, which is different from the conventional self-attention, as shown in Figure 2. Inspired by [7], the geometric distance among lane centerlines can serve as the basis for global dependencies, thereby enhancing the accuracy of lane topology reasoning. Therefore, we introduce the spatial proximity matrix (SPM) [44] to describe the normalized inverse geometric distances among lanes. It can be formulated as:

$$A_{SPM} = \text{Norm}\left(\frac{1}{d(\hat{v}_{i,l}^p, \hat{v}_{j,0}^p) + \epsilon} \right), i, j \in N, \tag{2}$$

where $\hat{v}_{i,l}^p$ is the end point of predicted centerline $\hat{v}_i^p$, and $\hat{v}_{j,0}^p$ is the start point of predicted $\hat{v}_j^p$. ϵ is a small constant to avoid infinity, $d(\cdot)$ denotes the distance (*i.e.*, ℓ_1 distance), and Norm is a normalization operation that divides each entry in the A_{SPM} by the mean inverse distance. Our core idea is to aggregate the global information of the lanes based on their spatial distances. Therefore, as shown in Figure 2(b), we add A_{SPM} to the self-attention, the GSA can be formulated as the follows in the layer l:

$$\text{GSA}(X^l) = A^l \cdot X^l W_V^l \tag{3}$$

where

$$A^l = \text{softmax}\left(\frac{X^l W_Q^l \cdot (X^l W_K^l)^\top}{\sqrt{d}} + A_{SPM} \right), \tag{4}$$

where $W_Q^l, W_K^l, W_V^l \in \mathbb{R}^{d \times d}$ are the weights of linear layer, $\cdot$ denotes the matrix multiplication, $A^l \in \mathbb{R}^{N \times N}$ denotes the attention weights in the l^{th} lane aggeration layer, d and N denoted the hidden dimension and number of queries. Subsequently, the lane feature X^{l+1} passed into the feed-forward network (FFN) preceded and succeeded by residual connections and normalization layers, and the output of $(l+1)$-th LAL can be formulated as:

$$X^{l+1} = \text{Norm}(X^l + \text{FFN}(\text{GSA}(X^l))), \tag{5}$$

It is noteworthy that different from TopoMLP [37] which introduces position encoding to enhance the features of each lane centerline, our proposed LAL explicitly utilizes position encoding for the aggregation of global information (*i.e.,* spatial interaction among lanes), to exhibit stronger generalization.

3.3. Counterfactual Intervention Layer

The Counterfactual Intervention Layer (CIL) is designed to capture the reasonable road structure among lanes in the traffic scene which is a transformer-encoder-like layer with the core component of Counterfactual Self-Attention (CSA). According to the analysis by [7], geometry-based aggregation heavily relies on the detected lane centerlines $\hat{\mathcal{V}}^p$ in Eq.(1). Inaccuracies in centerline detection can interfere with the quality of the A_{SPM} and lead to erroneous relationship predictions. We propose to leverage lane feature self-attention weights to represent the learned road structure and utilize counterfactual road structures (*e.g.*, zero attention) to improve the learning of traffic scene structure.

Counterfactual Self-Attention is designed within the layer to predict relationships among lanes. Drawing inspiration from causal inference methodologies [28], we propose a counterfactual intervention to explore the effects of the learned attention weights. Specifically, we perform the counterfactual intervention $do(A = \overline{A})$ by creating a hypothetical attention weight $\overline{A}$ to replace the original, while maintaining the lane feature X and A_{SPM} unchanged. The structure of the counterfactual self-attention closely resembles that of the geometry-guided self-attention. However,

the primary distinction lies in the configuration of the attention weights:

$$\text{CSA}(X^l) = \text{softmax}\left(\overline{A^l} + A_{SPM}\right) \cdot X^l W_V^l, \quad (6)$$

where

$$\overline{A^l} = \text{Zeros}\left(\frac{X^l W_Q^l \cdot (X^l W_K^l)^\top}{\sqrt{d}}\right), \quad (7)$$

where $W_Q^l, W_K^l, W_V^l \in \mathbb{R}^{d\times d}$ are the weights of linear layer, Zeros denotes the operation of generating a zero matrix with the same shape as the original matrix. $\overline{A^l}$ represents the hypothetical attention weight at the l^{th} layer. The output of $(l+1)$-th CIL can be formulated as:

$$X^{l+1} = \text{Norm}(X^l + \text{FFN}(\text{CSA}(X^l))), \quad (8)$$

where X^l denotes the output of the l-th LAL, and the FFN has the same structure but operates independently, as described in Eq. 5.

3.4. Edge Prediction Head

In the graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, $\mathcal{E}$ means the set of relationships or edges between all node pairs $\{[v_i, v_j] \mid v_i, v_j \in \mathcal{V}, i \neq j\}$, where each edge is represented as a binary value $\{0, 1\}$ indicating whether there is a connection from v_i to v_j. To obtain the predicted $\hat{\mathcal{E}}$, we developed an edge prediction head. Specifically, given the final output lane feature $\widetilde{X}$ from our TopoFormer, the edge prediction head first applies two independent three-layer MLPs for the start lane and end lane:

$$\widetilde{X}_s' = \text{MLP}_s(\widetilde{X}), \quad \widetilde{X}_e' = \text{MLP}_e(\widetilde{X}), \quad (9)$$

where the subscript s and e represent the start lane and the end lane, respectively. MLP_s and MLP_e respectively denote the corresponding MLPs for lanes. For each pair of lane $\widetilde{x}_s' \in \widetilde{X}_s'$ and $\widetilde{x}_e' \in \widetilde{X}_e'$, the confidence in their relationship is computed as follows:

$$\hat{\mathcal{E}}_{s,e} = \text{sigmoid}\Big(\text{MLP}_{\text{edge}}\big(\text{concat}(\widetilde{x_s}', \widetilde{x_e}')\big)\Big), s \neq e \quad (10)$$

where the "concat" operation combines the feature dimensions and the output dimension of MLP_{edge} is 1. The sigmoid function constrains this output to the range [0,1].

3.5. Training and Inference

During the training phase, our loss is divided into two parts, including the loss of node detection $\mathcal{L}_{\mathcal{V}}$ and the loss of edge prediction $\mathcal{L}_{\mathcal{E}}$. The detection loss for node $\mathcal{L}_{\mathcal{V}}$ is decomposed into a lane category $\mathcal{V}^c$ classification and a centerline $\mathcal{V}^p$ regression loss:

$$\mathcal{L}_{\mathcal{V}} = \lambda_{cls} \cdot \mathcal{L}_{cls} + \lambda_{reg} \cdot \mathcal{L}_{reg}, \quad (11)$$

where λ_{cls} and λ_{reg} are the coefficients. The classification loss $\mathcal{L}_{cls}$ is a Focal loss [18] and the regression loss λ_{reg} is an L1 loss. Note that the regression loss is calculated on the normalized 3D coordinates of the predicted ordered points list $\hat{\mathcal{V}}^p$. For the loss of edge prediction $\mathcal{L}_{\mathcal{E}}$, the total indirect effect of learned road structure on the prediction can be represented by the difference between the observed prediction $\hat{\mathcal{E}}_A = \hat{\mathcal{E}}(do(A = A), X = \widetilde{X})$ and its counterfactual result $\hat{\mathcal{E}}_{\overline{A}} = \hat{\mathcal{E}}(do(A = \overline{A}), X = \widetilde{X})$:

$$\hat{\mathcal{E}}_{\text{TIE}} = \mathbb{E}_{\overline{A}\sim\gamma}[\hat{\mathcal{E}}_A - \hat{\mathcal{E}}_{\overline{A}}], \quad (12)$$

where $\widetilde{X}$ denotes the final output lane feature, $\hat{\mathcal{E}}_{\text{TIE}}$ refers to the total indirect effect on the prediction, and γ is the distribution of the counterfactual attentions. Furthermore, we can adapt the focal loss on the $\hat{\mathcal{E}}_{\text{TIE}}$ with the coefficient λ_{cls}:

$$\mathcal{L}_{\mathcal{E}} = \lambda_{cls} \cdot \mathcal{L}_{\text{cls}}(\hat{\mathcal{E}}_{\text{TIE}}, \mathcal{E}_{\text{GT}}) \quad (13)$$

where $\mathcal{E}_{\text{GT}}$ denotes the ground truth of the edge and the classification loss $\mathcal{L}_{cls}$ is similar to Eq. (11).

Finally, our loss is the sum of the above losses:

$$\mathcal{L}_{total} = \mathcal{L}_{\mathcal{V}} + \mathcal{L}_{\mathcal{E}}. \quad (14)$$

During the inference phase, we no longer use $\hat{\mathcal{E}}_{\text{TIE}}$ as the predicted result for edges. Instead, we utilize $\hat{\mathcal{E}}_A$, which has not been subjected to counterfactual intervention.

3.6. Traffic topology reasoning

To validate the efficacy of our proposed T^2SG for downstream tasks, we selected the traffic topology reasoning task introduced by [33] for evaluation. The topology reasoning task involves detecting lane centerlines from multi-view images, 2D traffic elements from the front view image, and discerning relationships among them. The output of this task comprises two components: detection results and topology reasoning results. For the traffic elements detection, similar to the lane detection in Eq.(1), we get the multi-level 2D features $\mathcal{F}_{FV}$ obtained from the front view image processed through ResNet-50 [8] and FPN [17]. These features serve as the input for the Deformable Detector, represented as:

$$\begin{aligned} Q_{out}^T &= \text{DeformDETR}(Q_{init}^T, \mathcal{F}_{FV}), \\ \hat{T}^b, \hat{T}^c &= \text{TE Head}(Q_{out}^T), \end{aligned} \quad (15)$$

where $Q_{out}^T, Q_{init}^T \in \mathbb{R}^{N_{te}\times 256}$ represent the randomly initialized query and the output query from the last layer, respectively, where N_{te} denotes the number of queries. The elements $\hat{T}^b = [\hat{t}_i^b \in \mathbb{R}^4 \mid i = 1, 2, ..., N_{te}]$ and $\hat{T}^c = [\hat{t}_i^c \in \mathcal{C}_{te} \mid i = 1, 2, ..., N_{te}]$ correspond to the bounding box and class of the traffic elements, respectively. The TE Head is constructed by multilayer perceptron (MLPs). For lane detection, we employ the detection method of T^2SG.

The topology relationship results include the assignment of traffic elements to lane centerlines, denoted as $E_{lt} \in \mathbb{R}^{N \times N_{te}}$, and the connective relationships $E_{ll} \in \mathbb{R}^{N \times N}$ among lanes. Specifically, for E_{lt}, traffic elements and lanes that share the same class indicate connectivity. This connectivity is expressed as follows:

$$E_{lt}(i,j) = \begin{cases} 1, & \text{if } \hat{v}_i^c = \hat{t}_j^c \\ 0, & \text{otherwise.} \end{cases} \tag{16}$$

where $\hat{v}_i^c$ and $\hat{t}_j^c$ represent the categories of the lane $\hat{v}_i$ and traffic element $\hat{t}_j$, respectively, as obtained from Eq. (1) and Eq. (15). It is important to note that since our generated T^{2}SG focuses solely on the road itself, the category $\mathcal{C}_{lc}$ includes road signals, whereas $\mathcal{C}_{te}$ additionally encompasses traffic lights (i.e., red, green, and yellow). For the three types of traffic lights, we use two MLP layers to reduce the feature dimension for each lane and traffic light instance, and then the concatenated feature is sent into another MLP with a sigmoid activation to predict their relationship. For E_{ll}, we directly use the results of the edge $\hat{\mathcal{E}}$ from our T^{2}SG model as E_{ll}, denoted as $[E_{ll}(i,j) = \hat{e}_{ij} \mid \hat{e}_{ij} \in \hat{\mathcal{E}}, i \neq j]$.

4. Experiments

4.1. Datasets and Evaluation Setting

OpenlaneV2 Dataset. The OpenLane-V2 dataset [33] presents two unique subsets, $subset_A$ and $subset_B$, which are derived from the Argoverse 2 [36] and nuScenes [1] datasets, respectively. Each subset comprises 1,000 scenes. For the T^{2}SG generation task, we constructed a corresponding dataset based on OpenLane-V2, which contains 10 categories [1] of the lane (*i.e.*, $\mathcal{C}_{lc} = 10$). For the traffic topology reasoning task, we follow the same experimental setting in [33]. We conducted the Traffic Topology Reasoning Task training based on T^{2}SG and used the same MLP as in [13] to independently learn the bipartite graph between traffic lights and lanes.

Metric. For the T^{2}SG generation task, we adopt the Scene Graph Detection (SGDet) evaluation settings [10] and report the Average Precision (AP) of lane centerline detection which is class agnostic and mean Average Precision (mAP) that aggregates the AP for each category. Following [37], the detection employs the Fréchet distance for quantifying similarity and we report the AP and mAP for the match thresholds set at $\{1.0, 2.0, 3.0\}$. For the edge in the scene graph, we compute the accuracy (A@1) as the evaluation metric. For the topology reasoning task, we utilize the OpenLane-V2 [33] topology evaluation settings and report DET_l, DET_t, TOP_{ll} and TOP_{lt}, which are the mAP on lane centerlines, traffic elements, topology among lanes and topology between lanes and traffic elements, respectively. To summarize the overall effect of primary detection and topology reasoning, the OpenLane-V2 Score (OLS) is calculated as follows:

$$\text{OLS} = \frac{1}{4}\left[\text{DET}_l + \text{DET}_t + f(\text{TOP}_{ll}) + f(\text{TOP}_{lt})\right], \tag{17}$$

where f is the square root function.

[1] $\mathcal{C}_{lc}$= {lane, go_straight, turn_left, turn_right, no_left_turn, no_right_turn, u_turn, no_u_turn, slight_left, slight_right}

4.2. Implementation Details

We implement our model based on Pytorch and MMdetection on 16 Tesla V100 GPUs with a total batch size of 16. All images are resized into the same resolution of 1550 × 2048, and we use ResNet-50 [8] backbone pre-trained on ImageNet paired with a Feature Pyramid Network (FPN) [17] to extract multi-scale features. The dimension of hidden feature d is set to 256. The size of BEV grids is set to 200×100. For Lane centerline detection, the number of queries N is 200, and the number of points in centerlines l is 11. For the traffic elements detection, the number of queries is 100. The overall model TopoFormer is trained by AdamW optimizer with a weight decay of 0.01. The learning rate is initialized with 2e-4 and we employ a cosine annealing schedule [20] for the learning rate. Following [10], to accelerate convergence, we first train the lane detector and subsequently train the T^{2}SG task using the pre-trained lane detector.

4.3. Results

T^{2}SG generation. We present the quantitative performance of our proposed TopoFormer in T^{2}SG generation, comparing it with state-of-the-art scene graph generation methods in Table 1. The baseline reflects unprocessed input queries, while other methods focus on the semantic information related to traffic rules represented by the objects. Our approach emphasizes both semantic information and global lane dependencies, resulting in superior performance, In contrast, GCN-based methods like 3DSSG [31] and EdgeGCN [42] exhibit low accuracy in edge prediction due to global modeling constraints, while our method aggregates global lane features using the geometric information of centerlines, leading to improved results. Specifically, in terms of edge accuracy, GCN-based methods (e.g., 3DSSG) significantly lag behind Transformer-based methods (e.g., SGformer) with scores of 8.5 vs. 0.4 in $\text{A@1}_{1.0}$ and 34.6 vs. 4.7 in $\text{A@1}_{3.0}$. Furthermore, when simultaneously modeling semantic information and global dependencies, our proposed TopoFormer outperforms Transformer-based methods in both node and edge accuracy, demonstrating its effectiveness in traffic scene graph generation tasks.

Traffic topology reasoning. We present the quantitative performance of our TopoFormer in traffic topology reasoning in Table 2, our method surpasses other methods with

Method	Node						Edge		
	$AP_{1.0}$	$AP_{2.0}$	$AP_{3.0}$	$mAP_{1.0}$	$mAP_{2.0}$	$mAP_{3.0}$	$A@1_{1.0}$	$A@1_{2.0}$	$A@1_{3.0}$
Baseline	10.4	34.5	53.1	4.1	10.3	14.6	8.0	23.2	32.4
w/ 3DSSG [31]	10.7(+0.3)	36.1(+1.6)	54.2(+1.1)	4.4(+0.3)	10.3(+0.0)	15.1(+0.5)	0.4(-7.6)	2.0(-21.2)	4.7(-27.7)
w/ EdgeGCN [42]	10.8(+0.4)	36.6(+2.1)	54.5(+1.4)	4.5(+0.4)	10.3(+0.0)	15.8(+1.2)	0.4(-7.6)	2.1(-21.1)	5.4(-27.0)
w/ EGTR [10]	10.8(+0.4)	36.3(+1.8)	54.5(+1.4)	4.5(+0.4)	10.3(+0.0)	16.0(+1.4)	8.1(+0.1)	23.6(+0.4)	33.5(+1.1)
w/ SGformer [21]	11.2(+0.8)	36.8(+2.3)	**54.8(+1.7)**	4.6(+0.5)	10.9(+0.6)	16.5(+1.9)	8.5(+0.5)	24.8(+1.6)	34.6(+2.2)
w/ TopoFormer (Ours)	**11.8(+1.4)**	**37.5(+3.0)**	**54.8(+1.7)**	**4.8(+0.7)**	**11.3(+1.0)**	**16.7(+2.1)**	**8.8(+0.8)**	**25.6(+2.4)**	**35.6(+3.2)**

Table 1. **Comparisons of our model and existing state-of-the-art scene graph generation methods on OpenLane-V2** [33]. The subscripts represent Fréchet distance thresholds in the set of $\{1.0, 2.0, 3.0\}$. More details are described in Metrics. The best performances are highlighted in **bold**, while the second one is underlined. Red indicates the absolute improvements compared with the baseline, while blue indicates the decreases compared with the baseline.

Dataset	Method	Conference	$DET_l \uparrow$	$DET_t \uparrow$	$TOP_{ll} \uparrow$	$TOP_{lt} \uparrow$	OLS↑
$subset_A$	STSU [2]	ICCV2021	12.7	43.0	2.9	19.8	29.3
	VectorMapNet [19]	ICML2023	11.1	41.7	2.7	9.2	24.9
	MapTR [15]	ICLR2023	17.7	43.5	5.9	15.1	31.0
	TopoNet [13]	Arxiv2023	28.5	48.1	10.9	23.8	39.8
	TopoMLP [37]	ICLR2024	28.3	**49.5**	21.6	26.9	44.1
	TopoLogic [7]	NeurIPS2024	29.9	47.2	23.9	25.4	44.1
	TopoFormer(Ours)	-	**34.7(+4.8)**	48.2	**24.1(+0.2)**	**29.5(+3.6)**	**46.3(+2.2)**
$subset_B$	STSU [2]	ICCV2021	8.2	43.9	-	-	-
	VectorMapNet [19]	ICML2023	3.5	49.1	-	-	-
	MapTR [15]	ICLR2023	15.2	54.0	-	-	-
	TopoNet [13]	Arxiv2023	24.3	55.0	6.7	16.7	36.8
	TopoLogic [7]	NeurIPS2024	25.9	54.7	21.6	17.9	42.3
	TopoFormer(Ours)	-	**34.8(+8.9)**	**58.9(+3.9)**	**23.2(+1.6)**	**23.3(+5.4)**	**47.5(+5.2)**

Table 2. **Comparisons of our model and existing state-of-the-art methods on $subset_A$ and $subset_B$** [33]. "-" denotes the absence of relevant data. The best performances are highlighted in **bold**, while the second one is underlined. Red indicates the absolute improvements compared with the second one.

a 46.3% OLS in $subset_A$. Compared with TopoNet [13], a graph-based method, our method achieved higher score in the topology reasoning task (24.1 *v.s.* 10.9 on TOP_{ll}, 29.5 *v.s.* 23.8 on TOP_{lt}) while also achieves decent centerlines detection score (34.7 *v.s.* 28.5 on DET_l). This is attributed to T^2SG's ability to capture global lane information via LAL, enhancing topology reasoning performance. Compared to methods like TopoLogic [7], which also use geometric information, our TopoFormer improves DET_l and TOP_{lt} by effectively modeling road structure with CIL. Additionally, TopoFormer outperforms all models on $subset_B$, where centerlines are annotated in 2D space, demonstrating superior generalization.

4.4. Ablation Studies

Effects of the type of lane aggregation in LAL. To investigate the effectiveness of Geometry-guided Self-Attention, we introduced a variant labeled "w/o A_{SPM}", which excludes geometric information, along with three additional variations: "Add", "Mul", and "Had" representing the addition, multiplication, and Hadamard product of A_{SPM} with the self-attention weights, respectively. As shown in Table 3, we focus on the performance of these variants in the lane-lane relationship (*i.e.*, TOP_{ll}). The incorporation of A_{SPM} leads to superior performance across all variants compared to "w/o A_{SPM}", highlighting the importance of geometric information. Among all methods, the Add variant demonstrates the highest performance.

Effects of the type of counterfactual intervention in CIL. To investigate the effectiveness of counterfactual intervention, we introduced a variant labeled "w/o CIL", which excludes counterfactual intervention, along with three additional variations: "CIL-Zero", "CIL-Mean", "CIL-Random" representing the counterfactual interventions corresponding to zeros, the mean of attention weights, and randomly generated matrices, respectively. As shown in Ta-

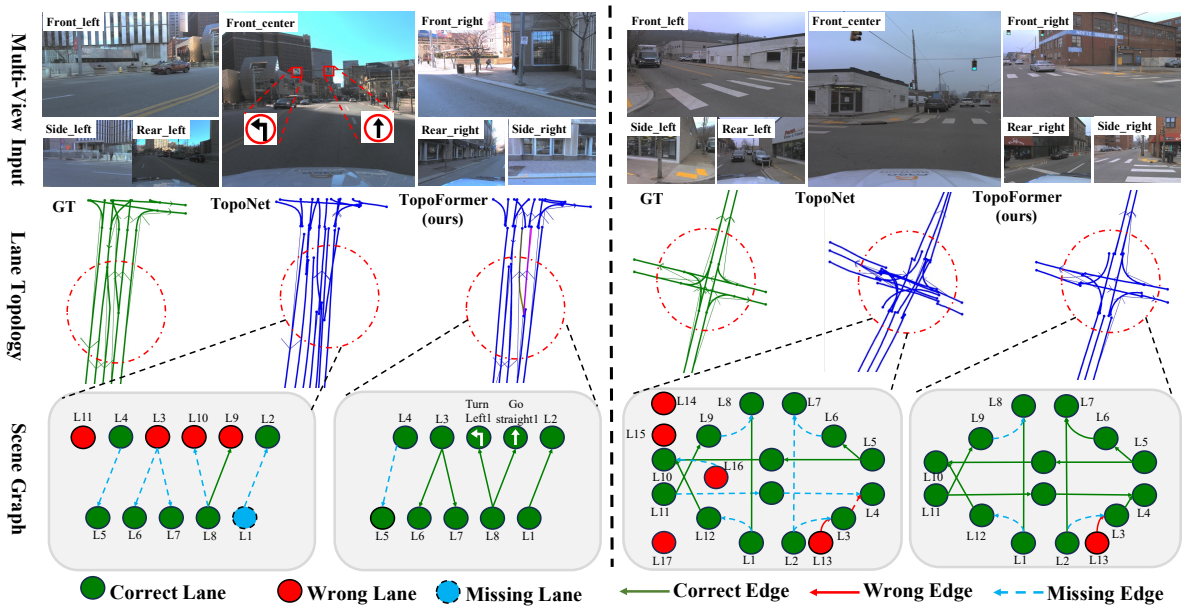


Figure 3. Qualitative results of the T^{2}SG generation task and the lane topology reasoning, comparing the performance of TopoNet [13] and our proposed TopoFormer. The first row represents multi-view inputs. The second row illustrates the results of lane detection and lane topology reasoning. The third row visualizes our defined T^{2}SG, with TopoNet's results converted to the same format for comparison. In these visualizations, green signifies correct predictions, red denotes erroneous predictions, and blue indicates missing predictions.

ble 3, all implementations yield significant improvements over the "w/o CIL", demonstrating the generality of the proposed CIL. Furthermore, 1) the "CIL-Zero" exhibits marginally superior performance compared to the others, potentially because the zero matrices represent a completely untrue road structure, while the mean and random matrices still contain some possible structures, which better accentuates the causal impact of reasonable road structure on topological relationships. 2) The performance of these implementations is closely matched, indicating the robustness of the counterfactual intervention implementation, which can adapt to various implementations.

4.5. Qualitative Analysis

Figure 3 presents a qualitative comparison between TopoNet [13] and our TopoFormer. The first row shows multi-view inputs of realistic scenes, while the second row displays lane topology results in the bird's eye view for both methods alongside the ground truth. The third row visualizes our defined T^{2}SG, with TopoNet's results converted to the same format for comparison. The results indicate that TopoFormer outperforms TopoNet in lane centerline detection and topology reasoning, accurately predicting most centerlines in both road structures (*i.e.*, straight and intersections). Additionally, our T^{2}SG not only captures the traffic topology structure but also categorizes each lane, effectively demonstrating the learning of road signal semantics.

Methods	DET$_l$	DET$_t$	TOP$_{ll}$	TOP$_{lt}$	OLS
w/o A_{SPM}	34.1	<u>47.3</u>	21.6	29.1	45.4
w/ Had A_{SPM}	<u>34.6</u>	46.4	22.0	<u>29.3</u>	45.5
w/ Mul A_{SPM}	**34.7**	46.3	<u>22.9</u>	29.0	<u>45.7</u>
w/ Add A_{SPM}	**34.7**	**48.2**	**24.1**	**29.5**	**46.3**
w/o CIL	32.2	47.0	22.2	28.6	44.9
CIL-Mean	<u>34.1</u>	<u>47.5</u>	21.1	28.4	45.2
CIL-Random	34.0	47.2	<u>22.9</u>	<u>28.8</u>	<u>45.6</u>
CIL-Zero	**34.7**	**48.2**	**24.1**	**29.5**	**46.3**

Table 3. Results of our TopoFormer with different variants on the traffic topology reasoning in OpenLane-V2 $subset_A$ set [33]. The best performances are highlighted in bold, while the second one is underlined. The gray shading part indicates we are more focused on TOP$_{ll}$.

5. Conclusion

In this paper, we introduced a new traffic topology scene graph (T^{2}SG) for traffic scene understanding and presented the TopoFormer, a one-stage topology scene graph transformer for T^{2}SG generation. TopoFormer features a Lane Aggregation Layer for global lane feature aggregation and a Counterfactual Intervention Layer to explore the road structure of the traffic scene. Our experiments demonstrate that TopoFormer outperforms state-of-the-art methods in T^{2}SG generation and significantly enhances traffic topology reasoning on the OpenLane-V2 benchmark.

6. Acknowledgement

This work is partly supported by the Funds for the National Natural Science Foundation of China under Grant 62202063 and U24B20176, Beijing Natural Science Foundation (L243027), Beijing Major Science and Technology Project under Contract No. Z231100007423014.

References

[1] Holger Caesar, Varun Bankiti, Alex H Lang et al. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 6

[2] Yigit Baran Can, Alexander Liniger, Danda Pani Paudel et al. Structured bird's-eye-view traffic scene understanding from onboard images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15661–15670, 2021. 2, 7

[3] Yigit Baran Can, Alexander Liniger, Danda Pani Paudel et al. Topology preserving local road network estimation from single onboard camera image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17263–17272, 2022. 2

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve et al. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 1, 3

[5] Sergio Casas, Abbas Sadat and Raquel Urtasun. Mp3: A unified model to map, perceive, predict and plan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14403–14412, 2021. 1

[6] Yuren Cong, Michael Ying Yang and Bodo Rosenhahn. Reltr: Relation transformer for scene graph generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9):11169–11183, 2023. 2

[7] Yanping Fu, Wenbin Liao, Xinyuan Liu et al. Topologic: An interpretable pipeline for lane topology reasoning on driving scenes. *Advances in Neural Information Processing Systems*, 37:61658–61676, 2024. 2, 4, 7

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren et al. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4, 5, 6

[9] Yuanmin Huang, Mi Zhang, Daizong Ding et al. Causalpc: Improving the robustness of point cloud classification by causal effect identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19779–19789, 2024. 3

[10] Jinbae Im, JeongYeon Nam, Nokyung Park et al. Egtr: Extracting graph from transformer for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24229–24238, 2024. 2, 6, 7

[11] Justin Johnson, Ranjay Krishna, Michael Stark et al. Image retrieval using scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3668–3678, 2015. 2

[12] Ranjay Krishna, Yuke Zhu, Oliver Groth et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73, 2017. 2

[13] Tianyu Li, Li Chen, Huijie Wang et al. Graph-based topology reasoning for driving scenes. *arXiv preprint arXiv:2304.05277*, 2023. 1, 4, 6, 7, 8

[14] Zhiqi Li, Wenhai Wang, Hongyang Li et al. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *European conference on computer vision*, pages 1–18. Springer, 2022. 4

[15] Bencheng Liao, Shaoyu Chen, Xinggang Wang et al. Maptr: Structured modeling and learning for online vectorized hd map construction. *arXiv preprint arXiv:2208.14437*, 2022. 1, 2, 7

[16] Bencheng Liao, Shaoyu Chen, Bo Jiang et al. Lane graph as path: Continuity-preserving path-wise modeling for online lane graph construction. In *European Conference on Computer Vision*, pages 334–351. Springer, 2025. 2

[17] Tsung-Yi Lin, Piotr Dollár, Ross Girshick et al. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 4, 5, 6

[18] Tsung-Yi Lin, Priya Goyal, Ross Girshick et al. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017. 5

[19] Yicheng Liu, Tianyuan Yuan, Yue Wang et al. Vectormapnet: End-to-end vectorized hd map learning. In *International Conference on Machine Learning*, pages 22352–22369. PMLR, 2023. 1, 2, 7

[20] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 6

[21] Changsheng Lv, Mengshi Qi, Xia Li et al. Sgformer: Semantic graph transformer for point cloud-based 3d scene graph generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4035–4043, 2024. 2, 7

[22] Changsheng Lv, Shuai Zhang, Yapeng Tian et al. Disentangled counterfactual learning for physical audiovisual commonsense reasoning. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[23] Mengshi Qi, Weijian Li, Zhengyuan Yang et al. Attentive relational networks for mapping images to scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3957–3966, 2019. 2

[24] Mengshi Qi, Yunhong Wang, Jie Qin et al. Stagnet: An attentive semantic rnn for group activity and individual action recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(2):549–565, 2019. 1

[25] Mengshi Qi, Yunhong Wang, Annan Li et al. Stc-gan: Spatio-temporally coupled generative adversarial networks for predictive scene parsing. *IEEE Transactions on Image Processing*, 29:5420–5430, 2020. 2

[26] Mengshi Qi, Jie Qin, Yi Yang et al. Semantics-aware spatial-temporal binaries for cross-modal video retrieval. *IEEE Transactions on Image Processing*, 30:2989–3004, 2021. 2

[27] Limeng Qiao, Wenjie Ding, Xi Qiu et al. End-to-end vectorized hd-map construction with piecewise bezier curve. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13218–13228, 2023. 2

[28] Yongming Rao, Guangyi Chen, Jiwen Lu et al. Counterfactual attention learning for fine-grained visual categorization and re-identification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 1025–1034, 2021. 3, 4

[29] Sayan Deb Sarkar, Ondrej Miksik, Marc Pollefeys et al. Sgaligner: 3d scene alignment with scene graphs. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21927–21937, 2023. 2

[30] Lucas Tabelini, Rodrigo Berriel, Thiago M Paixao et al. Keep your eyes on the lane: Real-time attention-guided lane detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 294–302, 2021. 1

[31] Johanna Wald, Armen Avetisyan, Nassir Navab et al. Rio: 3d object instance re-localization in changing indoor environments. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7658–7667, 2019. 6, 7

[32] Johanna Wald, Helisa Dhamo, Nassir Navab et al. Learning 3d semantic scene graphs from 3d indoor reconstructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3961–3970, 2020. 2

[33] Huijie Wang, Tianyu Li, Yang Li et al. Openlane-v2: A topology reasoning benchmark for unified 3d hd mapping. In *NeurIPS*, 2023. 1, 2, 3, 5, 6, 7, 8

[34] Yanan Wang, Michihiro Yasunaga, Hongyu Ren et al. Vqa-gnn: Reasoning with multimodal knowledge via graph neural networks for visual question answering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21582–21592, 2023. 2

[35] Zhechao Wang, Peirui Cheng, Minxing Chen et al. Drones help drones: A collaborative framework for multi-drone object trajectory prediction and beyond. *Advances in Neural Information Processing Systems*, 37:64604–64628, 2024. 1

[36] Benjamin Wilson, William Qi, Tanmay Agarwal et al. Argoverse 2: Next generation datasets for self-driving perception and forecasting. In *Advances in Neural Information Processing Systems*, 2023. 6

[37] Dongming Wu, Jiahao Chang, Fan Jia et al. Topomlp: A simple yet strong pipeline for driving topology reasoning. In *The Twelfth International Conference on Learning Representations*, 2024. 2, 4, 6, 7

[38] Danfei Xu, Yuke Zhu, Christopher B Choy et al. Scene graph generation by iterative message passing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5410–5419, 2017. 2

[39] Xu Yang, Yongliang Wu, Mingzhuo Yang et al. Exploring diverse in-context configurations for image captioning. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[40] Yiming Yang, Yueru Luo, Bingkun He et al. Topo2seq: Enhanced topology reasoning via topology sequence learning. *arXiv preprint arXiv:2502.08974*, 2025. 2

[41] Guangyao Zhai, Evin Pınar Örnek, Shun-Cheng Wu et al. Commonscenes: Generating commonsense 3d indoor scenes with scene graphs. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[42] Chaoyi Zhang, Jianhui Yu, Yang Song et al. Exploiting edge-oriented reasoning for 3d point-based scene graph analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9705–9715, 2021. 6, 7

[43] Zhixin Zhang, Yiyuan Zhang, Xiaohan Ding et al. Online vectorized hd map construction using geometry. In *European Conference on Computer Vision*, pages 73–90. Springer, 2024. 2

[44] Lichen Zhao, Daigang Cai, Lu Sheng et al. 3dvg-transformer: Relation modeling for visual grounding on point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2928–2937, 2021. 4

[45] Pengfei Zhu, Mengshi Qi, Xia Li et al. Unsupervised self-driving attention prediction via uncertainty mining and knowledge embedding. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8558–8568, 2023. 1

[46] Xizhou Zhu, Weijie Su, Lewei Lu et al. Deformable detr: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations*, 2021. 4