Enhanced Motion Forecasting with Visual Relation Reasoning

Sungjune Kim[1], Hadam Baek[1], Seunggwan Lee[1], Hyung-gun Chi[2], Hyerin Lim[3], Jinkyu Kim[1*], and Sangpil Kim[1*]

[1]Korea University [2]Purdue University [3]Hyundai Motor Company

Abstract. In this work, we emphasize and demonstrate the importance of visual relation learning for motion forecasting task in autonomous driving (AD). Since exploiting the benefits of RGB images in the existing vision-based joint perception and prediction (PnP) networks is limited in the perception stage, we delve into how the explicit utilization of the visual semantics in motion forecasting can enhance its performance. Specifically, this work proposes **ViRR**(**Vi**sual **R**elation **R**easoning), which aims to provide the prediction module with complex visual reasoning of relationships among scene agents. To achieve this, we construct a novel visual scene graph, where the pairwise visual relations are first aggregated as each agent's node feature. Then, the relations of the nodes are learned via higher-order relation reasoning method, which leverages the consecutive powers of the graph adjacency matrix. As a result, the extracted complex visual interrelations between the scene agents enable precise forecasting and provide explainable reasons for the model prediction. The proposed module is fully differentiable and thus can be easily applied to any existing vision-based PnP networks. We evaluate the motion forecasting performance of ViRR with challenging nuScenes benchmark and demonstrate its high necessity.

Keywords: Autonomous Driving · Motion Forecasting

1 Introduction

It is a recent trend in the field of autonomous driving (AD) to jointly optimize the perception and prediction sub-tasks in an end-to-end manner [10,24,35]. The idea is to perform prediction based on the outputs of the perception stage and to optimize the whole parameters with a single back-propagation. Especially, vision-based approaches are gaining attention for several reasons: (1) They provide rich contextual information that can be easily interpreted by humans; (2) the cost of deployment is more efficient compared to LiDAR or RADAR sensors.

However, the usage of the visual features in the existing vision-based perception and prediction (PnP) methods [10, 15] is limited in the perception stage. They mainly follow the Transformer [34] based detection and tracking architectures and abstract the visual perceptions in a query form, which operates as the

[*] Corresponding authors ({jinkyukim, spk7}@korea.ac.kr)

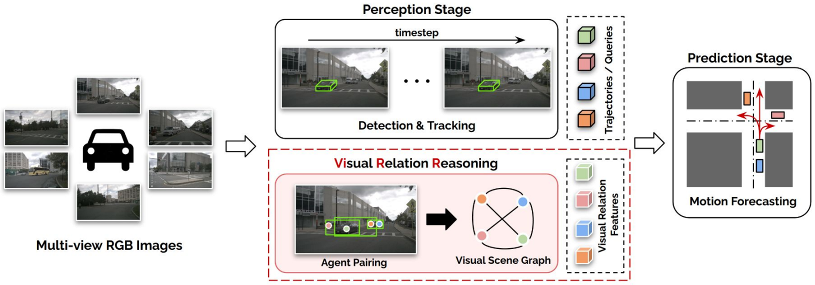

Fig. 1: Our proposed ViRR (Visual Relation Reasoning) can be flexibly attached to the existing vision-based joint perception and prediction pipelines. ViRR reasons about the complex visual relations of the road agents, which we prove is crucial for accurate and precise motion forecasting. Best viewed in color.

core component of the PnP pipeline. Although simple and informative, we argue that these queries should also contain the visual interrelations of the agents for enhanced downstream motion forecasting task.

Understanding the relations of the road agents in the visual space is crucial for motion forecasting in AD for a couple of reasons. First, the motion or the trajectories of the agents are easily susceptible to external factors, and the interactions between them are undoubtedly the most dominant ones [13]. Therefore, interpreting the visual relations (e.g. *next to*, *behind*, and *in front of*) can provide the model with detailed structures of the road environment. Furthermore, just as how humans predict the future motions, training the model to learn the visual relations of the surroundings can produce more reasonable predictions with increased explainability.

Therefore, we propose **ViRR** (**Vi**sual **R**elation **R**easoning), which learns the complex visual relations between diverse road agents. First, we reformulate the visual scene representation in a graph structure. Since graphs are powerful at relational modeling, it is a strategic choice to model the dynamic connections in the scene as a unified graph network [36,43]. Building upon the established scene graph generation benchmarks [33, 38], we extract the pairwise relation features between the identified agents using the RoIAlign [11] technique. However, our method is innovative in that instead of converting the pairwise relation features as pre-defined edge types, we aggregate these features for each agent, transforming them into a cohesive agent node feature. As a result of this, each agent node in our graph summarizes the local relational semantics and interactions of the agents within the visual space. Furthermore, this transformation frees the model from the constraints of the pre-defined edge types while flexibly learning new visual relations that are relevant in the AD systems.

The relations between these node features are then learned through the higher-order relation reasoning technique [19]. The motivation behind this is that human decisions are affected not only by direct and nearing influences but also by socially distant ones. For example, it is a common case in the urban traffic scenario where one pair of interactions affects the nearby agents, and the effect ripples throughout the whole scene. We implement this chaining dynamics by leveraging the consecutive powers of the graph adjacency matrix, which is defined as the reciprocal of the agent distances in the 3D coordinates. Exploiting the fact that the k-th power of the adjacency matrix represents the number of paths of length k between two vertices, we leverage a linear combination method that fuses the node features from diverse degrees in a single graph layer operation.

The module is designed to be fully differentiable, enabling a flexible fusion with the existing PnP methods. Figure 1 illustrates the general concept of our proposed method. Using the challenging nuScenes [2] benchmark, we demonstrate the benefits of ViRR by applying it to the existing models. We provide an in-depth analysis of how our proposed model can significantly enhance motion forecasting performance both quantitatively and qualitatively, demonstrating the importance of visual understanding for motion forecasting.

In summary, the main contributions of this work are listed as follows:

- We demonstrate the importance of explicit utilization of visual semantics in motion forecasting, which has been overlooked in the existing vision-based joint perception and prediction methods.
- We propose an innovative visual scene graph architecture that extracts pairwise visual relations of road agents and learns higher-order connectivity in the visual space of autonomous vehicles.
- Our method enhances the motion forecasting performance and provides a solid baseline for further research on the visual understanding for motion forecasting.

2 Related Work

2.1 Joint Perception and Prediction

Joint optimization of perception and prediction (PnP) stands as a pivotal area of research in the recent field of autonomous driving. Unlike the standalone modules [26,29,31,42] or multi-task approaches [3,28,30], joint PnP has the advantage of alleviating uncertainties and error propagation. Early works such as FaF [24] and IntentNet [4] use LiDAR sensor data as input modality and demonstrate the benefits of joint optimization over standalone modules using convolutional neural networks. PnPNet [22] brings tracking into the loop and takes advantage of the temporal contexts. FIERY [14] first introduces an end-to-end prediction module using bird's-eye view from surrounding monocular cameras. Recent works such as ViP3D [10] and UniAD [15] are also vision-based PnP methods, that leverage queries as the principle component of the pipeline, which makes them fully differentiable. The main goal of this work is to enhance the motion forecasting performance in these vision-based PnP models, by reasoning about the agent relations in the visual space.

2.2 Scene Graph Generation

Scene Graph Generation (SGG) is gaining significant attention in the computer vision field for its potential in various visual reasoning tasks [5, 6, 16, 17, 21, 39]. Xu *et al* [37] first introduce an end-to-end model that generates a structured scene representation from an input image. Zellers *et al.* [40] analyze the role of motifs in SGG. Then, Tang *et al.* [33] address the bias issue in SGG and use a total direct effect to resolve the problem. Recently in SQUAT [18], the authors introduce a quad attention module, which selectively utilizes valid edges to remove uncertainties from invalid edges. These methods commonly score the pairwise visual relations and assign a relation type within pre-defined predicate labels. However, we transform this process and preserve the rich visual relation features, freeing the model from the predicate constraints.

3 Preliminaries

3.1 Vision-based Perception Stage

In our work, we follow the perception stage of the mainstream vision-based joint PnP methods and extract the following information for our visual relation reasoning. Note that for simplicity, the notations throughout this work are based on a single-view image only, unless otherwise stated.

Multi-view Image Feature Maps. In each scene, a set of images $\mathcal{I} = \{\mathcal{I}_t\}_{t=-T}^{0}$ containing the T previous sequence of images up to the current timestamp are fed as the perception stage input. Then, the convolutional neural network (e.g. ResNet-50 [12]), followed by the feature pyramid network [23] extracts the 2D image feature maps $F = \{F_t\}_{t=-T}^{0}$ for each image in $\mathcal{I}$. For our visual relation reasoning, we utilize the feature maps of the current timestamp F_0, and further denote it as F for simplicity.

3D Bounding Boxes. The detection heads in the perception stage identify the road agents at each timestamp t. These results are annotated as 3D bounding boxes, which can be denoted as $\hat{\mathcal{B}} = \{\{\hat{\mathcal{B}}_t^n\}_{n=1}^{N_t}\}_{t=-T}^{0}$, where N_t is the total number of detected agents at timestamp t. Then, these agents are traced by the multi-agent object tracking modules throughout the observed image frames and we denote the 3D bounding boxes of the final N tracked agents as $\hat{\mathcal{B}} = \{\hat{\mathcal{B}}^n\}_{n=1}^{N}$.

3.2 Graph Convolutional Networks

Given a graph $G = (V, E)$ with V and E as a set of nodes and edges, respectively, a single layer of graph convolutional network [20] can be formulated as follows:

$$\mathbf{X}^{(l+1)} = \sigma(\hat{\mathbf{A}}\mathbf{X}^{(l)}\mathbf{W}^{(l)}), \tag{1}$$

where $\mathbf{X} \in \mathbb{R}^{|V| \times d}$ is a d-dimensional feature matrix of the nodes, $\hat{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{A}+\mathbf{I})\mathbf{D}^{-\frac{1}{2}}$ is a symmetrically normalized adjacency matrix of $\mathbf{A}$, $\mathbf{W} \in \mathbb{R}^{d \times d'}$ is a trainable weight matrix and $\sigma(\cdot)$ is an activation function. ViRR adopts this

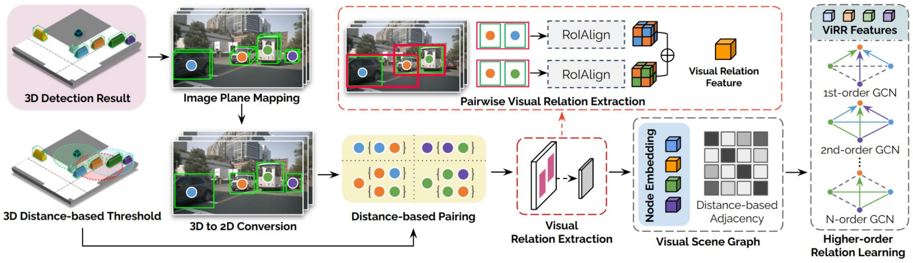

Fig. 2: The overall architecture of ViRR. First, we take the 3D bounding boxes and convert them into 2D. Then we pair the agents based on the 3D distance threshold, and extract the pairwise visual relation features, which we use as the node features of the graph. Lastly, the relations of these nodes are learned with our higher-order relation reasoning method. Best viewed in color.

concept to learn the visual relational semantics of the road agents. The specific design of the feature matrix, the adjacency matrix, and the higher-order message passing mechanism will be introduced in the following section.

4 ViRR: Visual Relation Reasoning

In this section, we explain our proposed ViRR in more detail. From a broad perspective, ViRR is a graph learning paradigm. Therefore, we first introduce a pairwise visual relation extraction method, where we obtain the node features. Then, we explain our higher-order relation learning method and elaborate on how the higher-order visual reasoning is achieved in the graph. The overall process is illustrated in Figure 2.

4.1 Visual Relation Extraction

3D to 2D Image Plane Mapping. We locate the 3D bounding box coordinates onto the corresponding image view using camera parameters. Then, we convert the 3D boxes into 2D, where lines are drawn to tightly fit the 3D box in the image plane. The 2D bounding boxes for each agent in the image are denoted as $\mathcal{B} = \{\mathcal{B}^m\}_{m=1}^{M}$, where M is the total number of agents in the camera view.
3D Distance-based Pairing. We set a distance threshold θ based on the center points of the agents' 3D bounding boxes to identify pairs that have the potential to influence the motion of one another. Then, within the camera view, we find the agent pairs of which we extract the visual relation features. In a multi-view camera setting, it is possible that the pair can be found in a different camera view. Even though the visual relations couldn't be captured in these cases, our graph adjacency is designed to connect the whole surrounding scenes based on 3D distance, so that the relations of the nearby agents in different camera views can all be taken into account. This will be explained in more detail in Sec 4.2.

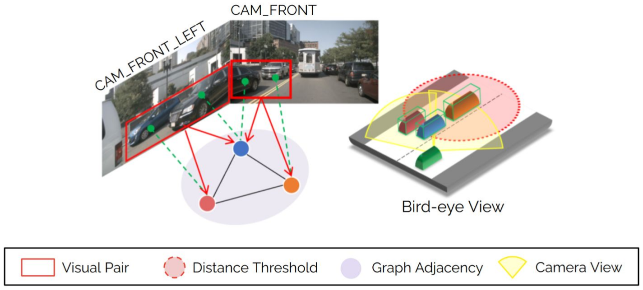

Fig. 3: The visual relations are captured between the agents within the same camera view, but this information is designed to propagate beyond the camera views in our graph. For example, the red and blue agents are in the threshold radius of the orange agent. Even though the red agent is in a different camera view from the orange agent, the overall relations are shared in the graph. Best viewed in color.

Pairwise Visual Relation Extraction. Motivated by the relation learning methods in the existing scene graph benchmark [33], we utilize the RoIAlign [11] technique to extract the pairwise visual relation features. In scene graph benchmarks, these features pass a softmax function and are converted to pre-defined edge types between the pair. However, we introduce a novel twist, where we aggregate the pairwise features per agent, transforming them into the agent node feature. For example, the aggregated visual relation feature $\mathbf{v}$ of a single agent n can be formulated as:

$$\mathbf{v}_n = \frac{\sum_{m \in \mathcal{N}(n)} \mathcal{R}(F, \mathcal{B}^n, \mathcal{B}^m)}{|\mathcal{N}(n)|}, \tag{2}$$

where $\mathcal{R}$ is the RoIAlign function, which takes the image feature maps F and the bounding box pair as its inputs, and $\mathcal{N}(n)$ denotes the paired neighbors of agent n. We take the average of the aggregated features by dividing the summed value with $|\mathcal{N}(n)|$, which is the total number of paired agents for agent n. For agents in the overlapping regions, we also take the average between the features from two different camera views.

Agent Node Feature. The extracted and aggregated features act as node features in our graph convolution. The initial feature $\mathbf{X}^{(0)}$ can be formulated as:

$$\mathbf{X}^{(0)} = f\left(\begin{bmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_N \end{bmatrix}\right) \in \mathbb{R}^{N \times d}, \tag{3}$$

where f is a linear mapping function, and d is the embedding dimension. The main advantage of this design is that these features encapsulate the rich local relations between the neighboring agents in a visual space. Also, our method

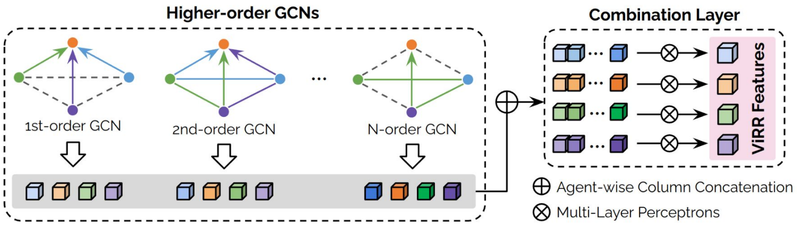

Fig. 4: The visualization of a single layer process of the higher-order relation learning. First, the multiple orders of GCNs aggregate the influences from the paths of various distances. Then the combination layer combines and distills meaningful features from these relations.

can preserve diverse visual relational semantics, unlike constraining them into pre-defined relation types.

4.2 Higher-order Visual Relation Learning

3D Distance-based Adjacency. In this work, the local pairwise visual features within each camera view are developed to propagate globally throughout the entire surrounding scene. This means that the information of the agents obtained in a single camera view can be shared across agents in different view points, enabling a panoramic understanding of the visual scene. The concept is visualized in Figure 3 for clarity. To do so, we construct an adjacency matrix $\mathbf{A} = \{a_{nm} | \forall n, m \in \{1, ..., N\}\}$, where the value of a_{nm} is the reciprocal of the distance between the two agents if $n \neq m$, and $d(n,m) < \theta$. Here, $d(\cdot,\cdot)$ denotes the distance between the pairs. Otherwise, the value is set as 0. Hence the pairs in further distance will have a lower weight, and the pairs with closer distance will have a bigger weight. Then, following the normalization technique described in section 3.2, we obtain the final adjacency matrix $\hat{\mathbf{A}}$.

Higher-order Graph Convolution. Taking the power of the adjacency matrix counts the number of walks of a certain length between pairs of nodes in the graph. We exploit this fact to model higher-order influence between the agents in the scene. First, we make P adjacency matrices in each layer, each of which represents the p^{th}-powers of the adjacency matrix $\hat{\mathbf{A}}$, where $p \in \{1, ..., P\}$. Graph convolution of each matrix with the agent visual relation feature $\mathbf{X}$ computes the p^{th}-order visual influences between the agents:

$$\mathbf{H}_p^{(l)} = \sigma(\hat{\mathbf{A}}^p \mathbf{X}^{(l)} \mathbf{W}_p^{(l)}), \tag{4}$$

where l is a layer number and $\mathbf{W}_p^{(l)}$ is a trainable weight that learns the p^{th}-order relations in layer l.

Multi-order Feature Combination. We introduce a combination layer, which combines and distills meaningful influences from multiple orders of relations, and pass them to the next graph layer. First, the combination layer concatenates the P higher-order features. Next, it goes through the mixture function, where we utilize multi-layer perceptrons (MLPs) with a non-linear activation function. The described procedure can be formulated as follows:

$$\mathbf{X}^{(l+1)} = \sigma(\mathrm{MLP}(\lambda_1 \mathbf{H}_1^{(l)} \parallel \lambda_2 \mathbf{H}_2^{(l)} \parallel ... \parallel \lambda_P \mathbf{H}_P^{(l)})), \tag{5}$$

where λ_p denotes the trainable scalar weights for the p^{th}-order feature, and $\parallel$ denotes column concatenation. We let $\sum_{p\in\{1,...,P\}} \lambda_p = 1$, so that the layer weighs the influences from different orders. We stack the explained graph convolution and combination layers repeatedly and obtain the final ViRR features for each agent, which are combined with the initial perception stage outputs. The described process is visualized in Figure 4.

5 Experimental Settings

5.1 Dataset

We use the nuScenes [2] dataset, which is widely employed as a benchmark for autonomous driving in the motion forecasting task. It consists of 1000 driving scenes, each of which lasts for 20 seconds. The data is collected in Boston and Singapore using various types of sensors including RGB cameras, LiDAR and RADAR. Also, it contains 1.4M bounding boxes that exist within the sampled 40k key-frames at 2Hz. Since the dataset has a 360-degree field of view captured from six cameras, it serves a pivotal role in vision-based PnP methods.

5.2 Evaluation Metrics

Motion Forecasting. We use minimum of Average Displacement Error (minADE), minimum of Final Displacement Error (minFDE), and Miss Rate (MR) as the evaluation metrics for motion forecasting. minADE measures the minimum of the average L2 distances between the ground truth and the predicted trajectories. Similarly, minFDE measures the minimum of the L2 distances between the ground truth and the predicted trajectories at the final timestamp. MR represents the ratio of minFDE values that exceed certain a threshold.
Detection & Tracking. We measure the detection performance with Average Precision (AP). Tracking performance is reported with Average Multi-object Tracking Accuracy (AMOTA), Average Multi-object Tracking Precision (AMOTP), and Identity Switches (IDS).

5.3 Baselines

We examine our method with popular vision-based PnP methods ViP3D [10] and UniAD [15], both of which utilize the concept of queries as the core thread throughout the pipeline. In UniAD, we conduct experiments with two different types of PnP model variation. One jointly optimizes the tracking and motion

Table 1: The motion forecasting performance enhancements with our proposed ViRR. Our method noticeably improves the previous baselines, demonstrating the importance of visual relation reasoning for motion forecasting. The abbreviations `T`, `M` and `Mo` in UniAD rows indicate Tracking, Map and Motion Forecasting, respectively. The performance improvement achieved by our proposed method is indicated in **bold**.

	w/o ViRR			w/ ViRR		
	minADE ↓	minFDE ↓	MR ↓	minADE ↓	minFDE ↓	MR ↓
ViP3D [10]	2.051	2.862	0.244	1.690 (**+17.60%**)	2.075 (**+27.50%**)	0.191 (**+21.72%**)
UniAD (`T`+`Mo`) [15]	0.749	1.101	0.161	0.684 (**+8.68%**)	0.901 (**+18.17%**)	0.051 (**+68.32%**)
UniAD (`T`+`M`+`Mo`) [15]	0.732	1.063	0.158	0.628 (**+14.20%**)	0.891 (**+16.21%**)	0.040 (**+74.55%**)
MOTR + CVAE Motion	0.976	1.281	0.188	0.951 (**+2.47%**)	1.381 (**+16.16%**)	0.166 (**+11.68%**)

forecasting tasks, and the other additionally optimizes map segmentation in the perception stage. Furthermore, we provide a simple vision-based joint perception and prediction (PnP) baseline model to demonstrate the versatility of our proposed method. Specifically, we adopt the perception stage algorithm from MOTR [41] and implement the motion forecasting stage with CVAE [32] algorithm. The CVAE motion decoder mainly consists of three different parts. First, the Recurrent Neural Network (RNN)-based model encodes temporal information of the ground truth future trajectory. This process is only operated in the training step. Second, the CVAE Network transforms the agent features and the encoded ground truth future trajectory features into the probability distributions. The divergence between the two probability distributions is minimized in the training step. This enables the agent features to be guided with the ground truth semantics in the inference step, even without the direct use of the ground truth features. Lastly, the Autoregressive RNN decoder predicts the future trajectories based on the agent features and the latent variables. The details of this architecture are explained in our supplementary material. In all of these baseline models, ViRR resides in between the perception and prediction stages and provides rich visual semantics that enhance the performance of motion forecasting.

5.4 Implementation Details

Our method is implemented with PyTorch [27] and MMCV [7] code base. Since ViRR is an intermediate module between the perception and prediction stages, we mostly follow the hyperparameters from the existing baselines. To solely analyze the performance of motion forecasting, we pre-train the perception stage and freeze it, thereby only training the parameters of ViRR and motion forecasting. However, the result of joint optimization is also provided in Sec. 6.1. We empirically find the optimal value for the distance threshold θ in between $10m$ and $20m$ in the 3D coordinate. Following the convention, the spatial dimension of the output from the RoIAlign operation is set as (7×7). Lastly, the number of graph convolution layers and the higher-order observation degree per layer is determined in between $\{1, 2, 3\}$ and $\{1, 2, 4, 8\}$, respectively. Two layers of graph

Table 2: We examine the dependencies between the perception results and the influence of ViRR in motion forecasting. T denotes the number of observation frames. The prediction results are reported in the form of (w/o ViRR → w/ ViRR), and the relative improvements in percentage .

	$T = 1$		$T = 3$	
Metrics	Vehicles	Pedestrian	Vehicles	Pedestrian
AP ↑	0.339	0.461	0.340	0.464
AMOTA ↑	0.618	0.436	0.623	0.442
minADE ↓	0.75 → 0.65 (+13.33%)	0.81 → 0.72 (+11.11%)	0.73 → 0.62 (+15.07%)	0.78 → 0.69 (+11.54%)
minFDE ↓	1.10 → 0.95 (+13.64%)	1.11 → 0.97 (+12.61%)	1.06 → 0.88 (+16.98%)	1.09 → 0.93 (+14.68%)
MR ↓	0.15 → 0.06 (+60.00%)	0.13 → 0.05 (+61.54%)	0.16 → 0.04 (+75.00%)	0.13 → 0.04 (+69.23%)

convolutions and four higher-order degrees are used for our architecture which produces the best result.

6 Experimental Results

6.1 Quantitative Analysis

Prediction Performance Enhancement. As clearly indicated in Table 1, providing the model with visual relation semantics of the road agents with our ViRR consistently enhances the motion forecasting performance throughout all baselines. Specifically, ViRR significantly reduces the errors by 68.32% and 74.55% on MR compared to the UniAD methods. minADE and minFDE are also reduced by a large margin, especially in ViP3D where the metrics are improved by 17.60% and 27.50% respectively. Also, ViRR improves the metrics in our simple custom designed model, which demonstrates the adaptability of ViRR in diverse model environments. The results point out that ViRR is highly necessary for more precise motion forecasting.

Relation with Perception Results. We analyze the dependencies between the perception results and the effects of ViRR by comparing the two models with different numbers of observation frames in the perception stage. Typically, larger number of observation frames yields better perception performance. For each agent class, we first report the comparison of the perception results in the upper two rows in Table 2. Then, we relate these results with the effects of ViRR in motion forecasting. The results show that better perception enhances ViRR in both agent class through all metrics. Specifically, the improvement of the MR result of vehicles increased from 60.00% to 75.00% with better perception. We discover that the relative improvements that ViRR brings are largely correlated with the perception performance.

Table 3: We report the results of end-to-end joint optimization. The upper three rows indicate the tracking results of the UniAD model with tracking and motion forecasting optimization, and the lower three rows are its motion forecasting results.

Task	Metric	w/o ViRR	w/ ViRR
Tracking	AMOTA ↑	0.360	0.369 (+2.50%)
	AMOTP ↓	1.350	1.342 (+0.59%)
	IDS ↓	919	907 (+1.31%)
Motion Forecasting	minADE ↓	0.751	0.701 (+6.64%)
	minFDE ↓	1.109	0.954 (+13.98%)
	MR ↓	0.162	0.080 (+50.62%)

Joint PnP Optimization Results. So far, we have demonstrated the impact of ViRR in motion forecasting with frozen perception stages. In this part, we observe the impact of ViRR in an end-to-end joint PnP optimization manner. As shown in Table 3, the benefits of ViRR are also clear in the end-to-end optimization. Even though the relative improvements in motion forecasting in this experiment are smaller than the improvements in the models with pre-trained perception stages, ViRR solidly reduces the error metrics, producing closer predictions to the ground truth trajectories. More notably, applying ViRR also enhances the tracking performance, meaning visual representations of each agent and their relational semantics can benefit the entire end-to-end pipeline. We plan to explore deeper into this analysis in our future work.

6.2 Qualitative Analysis

In this section, we analyze the qualitative results of ViRR that the numerical results were unable to highlight. First, we present the comparisons of the trajectory predictions in the bird's-eye view. Then, we illustrate the agent features in the embedding space and show that our visual relational features contain semantics that the agent queries do not.

Prediction Performance Enhancement. We examine the scenarios where the improvements of the predictions with our ViRR are noticeable in Figure 5. In Figure 5-(a), the baseline predicts the vehicles on the right of the self-driving car to directly turn right with a high possibility, considering the upcoming corner. However, ViRR takes the geometrical and visual relations of the vehicles (e.g. the vehicle and a pedestrian on its front right in CAM_FRONT_RIGHT) into account, thereby avoiding the collision and producing reasonable trajectories. Figure 5-(b) highlights the benefits of our higher-order relation reasoning. The baseline predicts the vehicles on the left of the self-driving car to pass the crosswalk in the pedestrian signal. With the help of ViRR, the vehicle in the CAM_FRONT_LEFT view captures the crossing pedestrian and determines to

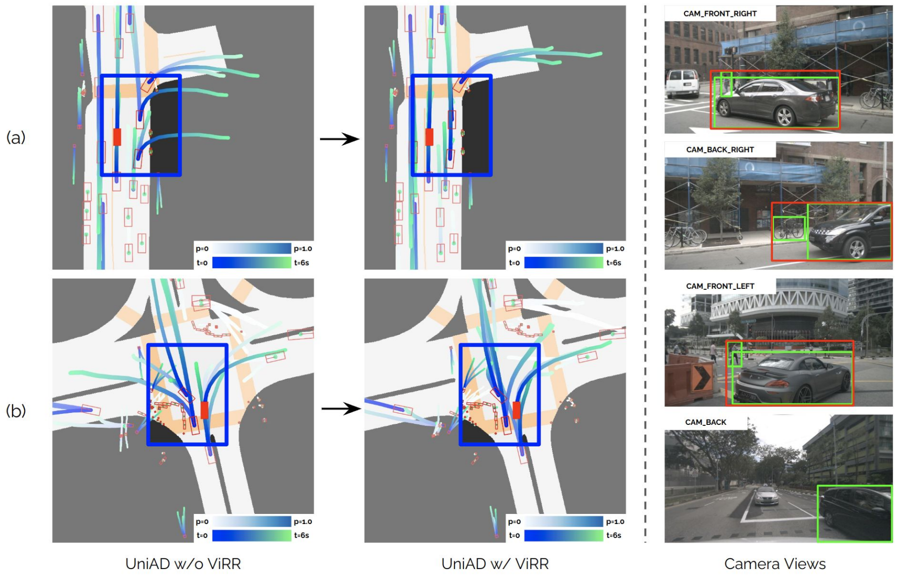

Fig. 5: Visualization of the qualitative benefits of our proposed ViRR. The first column is the result of the baseline method and the second column illustrates the enhanced result with ViRR. The last column is the corresponding camera views with important visual unions in a red rectangle. Best viewed in color.

stop. Furthermore, this first-order visual relation is conveyed to the vehicle in the CAM_BACK view, making it to avoid collision with the pedestrians.

Agent Feature Semantics Comparison. The visual relation reasoning provides the model with fruitful semantics that enable further clarification of the agent identity. To see how this works, the t-SNE [25] visualization of the comparison between the agent queries of the baseline and our ViRR agent features is provided in Figure 6. Here, we observe that the ViRR agent features form new types of clusters, where features gather based on the visual relations of the agents. This implies that ViRR can enrich the agent representations by extracting the semantics that the agent queries from the perception stage couldn't capture. In our work, the queries and the ViRR features are summed and fed as the prediction stage input, enhancing the motion forecasting of the agents.

6.3 Ablation Study

Types of Visual Semantic. Different types of visual semantics have different influences on the motion forecasting result. Most naively, we first provide each agent with the image feature of the camera scene that the agent is in. Then we move on to discover the impact of finer-drawn visual semantics, by providing

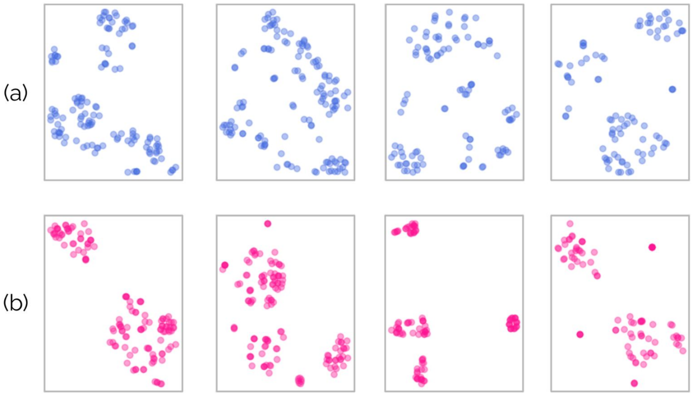

Fig. 6: t-SNE visualization of the comparison between two types of agent features. (a) is the agent queries from the perception stage of the baseline UniAD, and (b) is the agent features from our proposed ViRR. Best viewed in color.

Table 4: The comparison between different visual semantics provided for the prediction task. 'H.R.L.' denotes our proposed higher-order relation learning technique.

	minADE ↓	minFDE ↓	MR ↓
Baseline	0.732	1.063	0.158
Scene Feature	0.722	1.051	0.149
Agent BBox (ResNet-101 [12])	0.720	1.037	0.151
Agent BBox (ViT [9])	0.711	1.022	0.145
ViRR w/o H.R.L.	0.663	0.976	0.059
ViRR w/ H.R.L.	**0.628**	**0.891**	**0.040**

the agents with image features of their 2D bounding box. Here we compare the results between two image feature extractors ResNet-101 [12] and Vision Transformer [9], both of which are pre-trained on ImageNet1K [8]. Next, we use features from our developed visual relation extraction method. Lastly, we apply our higher-order visual relation learning technique and construct the whole ViRR architecture. The overall experimental procedure and the results in Table 4 derive the following observations: (1) Explicit utilization of the image features in the prediction stage improves the performance; (2) Finer-grained visual semantics are more beneficial in forecasting future motion; (3) Compared to individual agent visual features, pairwise visual relations and their higher-order reasoning are better at modeling social dynamics.

Types of Relation Learning. The relation learning of the pairwise visual semantics can be achieved through several different methods. We compare our

Table 5: The comparison between different relation learning methods. P denotes the number of higher-order observation degrees of our model.

	minADE ↓	minFDE ↓	MR ↓
Baseline	0.732	1.063	0.158
GCN [20]	0.683	0.987	0.073
MixHop [1]	0.664	1.002	0.055
Cross Attention [34]	0.655	0.929	0.049
ViRR ($P = 2$)	0.657	0.969	0.051
ViRR ($P = 4$)	**0.628**	**0.891**	**0.040**
ViRR ($P = 8$)	0.634	0.933	0.049

proposed higher-order relation learning method with different observation degrees against GCN [20], MixHop [1], and Cross-Attention [34]. The results are reported in Table 5. First, we discover that visual relation reasoning generally enhances the prediction regardless of the relation learning type. Next, comparing our result with a similar higher-order graph-based method MixHop, we show that our combination technique has a clear advantage over the conventional method. Lastly however, a larger observation degree does not imply better performance, meaning that the optimal value differs depending on the dataset, where $P = 4$ produced the best result in our experimental setting.

7 Conclusion

In this work, we study how visual relation reasoning enhances motion forecasting in autonomous driving. To this end, we develop ViRR, which extracts complex visual relations from pairwise agents and represents them in a novel scene graph. In contrast to the conventional scene graph approaches that convert visual relations as pre-defined edge types, ViRR aggregates these relations per agent, transforming them into comprehensive agent node features. Furthermore, our method leverages the consecutive powers of the adjacency matrix, enabling higher-order relation learning of the nodes. The quantitative and qualitative results of our study demonstrate the effectiveness of employing visual relation reasoning in predicting the future motions of road agents.

Limitations and Future Work. This work mainly focuses on the influences of visual relation reasoning in motion forecasting task. However, as shown in 6.1, we have witnessed the potential of the benefits of visual relation reasoning in other tasks of the autonomous driving pipelines. Therefore, we plan to extend this work and study its impact on the entire end-to-end autonomous driving systems as future work.

Acknowledgements This research was supported by Autonomous Driving Center, Hyundai Motor Company R&D Division(40%), Culture, Sports and Tourism R&D

Program through the Korea Creative Content Agency grant funded by the Ministry of Culture, Sports and Tourism in 2024((International Collaborative Research and Global Talent Development for the Development of Copyright Management and Protection Technologies for Generative AI, RS-2024-00345025, 4%),(Research on neural watermark technology for copyright protection of generative AI 3D content, RS-2024-00348469, 25%)), Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT)(RS-2019-II190079, 1%), IITP under the Leading Generative AI Human Resources Development(IITP-2024-RS-2024-00397085, 10%) grant, IITP grant (RS-2022-II220043, Adaptive Personality for Intelligent Agents, 10% and IITP-2024-2020-0-01819, ICT Creative Consilience program, 5%), and Basic Science Research Program through the NRF funded by the Ministry of Education(NRF-2021R1A6A1A13044830, 5%).

References

1. Abu-El-Haija, S., Perozzi, B., Kapoor, A., Alipourfard, N., Lerman, K., Harutyunyan, H., Ver Steeg, G., Galstyan, A.: Mixhop: Higher-order graph convolutional architectures via sparsified neighborhood mixing. In: international conference on machine learning. pp. 21–29. PMLR (2019)
2. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuscenes: A multimodal dataset for autonomous driving. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11621–11631 (2020)
3. Casas, S., Gulino, C., Suo, S., Luo, K., Liao, R., Urtasun, R.: Implicit latent variable model for scene-consistent motion forecasting. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIII 16. pp. 624–641. Springer (2020)
4. Casas, S., Luo, W., Urtasun, R.: Intentnet: Learning to predict intention from raw sensor data. In: Conference on Robot Learning. pp. 947–956. PMLR (2018)
5. Chang, X., Ren, P., Xu, P., Li, Z., Chen, X., Hauptmann, A.: A comprehensive survey of scene graphs: Generation and application. IEEE Transactions on Pattern Analysis and Machine Intelligence **45**(1), 1–26 (2021)
6. Chatterjee, M., Ahuja, N., Cherian, A.: Learning audio-visual dynamics using scene graphs for audio source separation. Advances in Neural Information Processing Systems **35**, 16975–16988 (2022)
7. Contributors, M.: MMCV: OpenMMLab computer vision foundation. `https://github.com/open-mmlab/mmcv` (2018)
8. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
9. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2020)
10. Gu, J., Hu, C., Zhang, T., Chen, X., Wang, Y., Wang, Y., Zhao, H.: Vip3d: End-to-end visual trajectory prediction via 3d agent queries. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5496–5506 (2023)
11. He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 2961–2969 (2017)

12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
13. Helbing, D., Molnar, P.: Social force model for pedestrian dynamics. Physical review E **51**(5), 4282 (1995)
14. Hu, A., Murez, Z., Mohan, N., Dudas, S., Hawke, J., Badrinarayanan, V., Cipolla, R., Kendall, A.: Fiery: Future instance prediction in bird's-eye view from surround monocular cameras. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 15273–15282 (2021)
15. Hu, Y., Yang, J., Chen, L., Li, K., Sima, C., Zhu, X., Chai, S., Du, S., Lin, T., Wang, W., et al.: Planning-oriented autonomous driving. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17853–17862 (2023)
16. Johnson, J., Gupta, A., Fei-Fei, L.: Image generation from scene graphs. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1219–1228 (2018)
17. Johnson, J., Krishna, R., Stark, M., Li, L.J., Shamma, D., Bernstein, M., Fei-Fei, L.: Image retrieval using scene graphs. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3668–3678 (2015)
18. Jung, D., Kim, S., Kim, W.H., Cho, M.: Devil's on the edges: Selective quad attention for scene graph generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18664–18674 (2023)
19. Kim, S., Chi, H.g., Lim, H., Ramani, K., Kim, J., Kim, S.: Higher-order relational reasoning for pedestrian trajectory prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15251–15260 (2024)
20. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: International Conference on Learning Representations (2016)
21. Li, X., Jiang, S.: Know more say less: Image captioning based on scene graphs. IEEE Transactions on Multimedia **21**(8), 2117–2130 (2019)
22. Liang, M., Yang, B., Zeng, W., Chen, Y., Hu, R., Casas, S., Urtasun, R.: Pnpnet: End-to-end perception and prediction with tracking in the loop. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 11553–11562 (2020)
23. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2117–2125 (2017)
24. Luo, W., Yang, B., Urtasun, R.: Fast and furious: Real time end-to-end 3d detection, tracking and motion forecasting with a single convolutional net. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 3569–3577 (2018)
25. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(11) (2008)
26. Nayakanti, N., Al-Rfou, R., Zhou, A., Goel, K., Refaat, K.S., Sapp, B.: Wayformer: Motion forecasting via simple & efficient attention networks. In: 2023 IEEE International Conference on Robotics and Automation (ICRA). pp. 2980–2987. IEEE (2023)
27. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. Advances in neural information processing systems **32** (2019)

28. Phillips, J., Martinez, J., Bârsan, I.A., Casas, S., Sadat, A., Urtasun, R.: Deep multi-task learning for joint localization, perception, and prediction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4679–4689 (2021)
29. Roh, W., Chang, G., Moon, S., Nam, G., Kim, C., Kim, Y., Kim, J., Kim, S.: Ora3d: Overlap region aware multi-view 3d object detection. arXiv preprint arXiv:2207.00865 (2022)
30. Sadat, A., Casas, S., Ren, M., Wu, X., Dhawan, P., Urtasun, R.: Perceive, predict, and plan: Safe motion planning through interpretable semantic representations. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIII 16. pp. 414–430. Springer (2020)
31. Simonelli, A., Bulo, S.R., Porzi, L., López-Antequera, M., Kontschieder, P.: Disentangling monocular 3d object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 1991–1999 (2019)
32. Sohn, K., Lee, H., Yan, X.: Learning structured output representation using deep conditional generative models. Advances in neural information processing systems **28** (2015)
33. Tang, K., Niu, Y., Huang, J., Shi, J., Zhang, H.: Unbiased scene graph generation from biased training. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3716–3725 (2020)
34. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
35. Wu, P., Chen, S., Metaxas, D.N.: Motionnet: Joint perception and motion prediction for autonomous driving based on bird's eye view maps. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11385–11395 (2020)
36. Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., Philip, S.Y.: A comprehensive survey on graph neural networks. IEEE transactions on neural networks and learning systems **32**(1), 4–24 (2020)
37. Xu, D., Zhu, Y., Choy, C.B., Fei-Fei, L.: Scene graph generation by iterative message passing. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5410–5419 (2017)
38. Yang, J., Lu, J., Lee, S., Batra, D., Parikh, D.: Graph r-cnn for scene graph generation. In: Proceedings of the European conference on computer vision (ECCV). pp. 670–685 (2018)
39. Yang, X., Tang, K., Zhang, H., Cai, J.: Auto-encoding scene graphs for image captioning. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10685–10694 (2019)
40. Zellers, R., Yatskar, M., Thomson, S., Choi, Y.: Neural motifs: Scene graph parsing with global context. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5831–5840 (2018)
41. Zeng, F., Dong, B., Zhang, Y., Wang, T., Zhang, X., Wei, Y.: Motr: End-to-end multiple-object tracking with transformer. In: European Conference on Computer Vision. pp. 659–675. Springer (2022)
42. Zhang, T., Chen, X., Wang, Y., Wang, Y., Zhao, H.: Mutr3d: A multi-camera tracking framework via 3d-to-2d queries. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4537–4546 (2022)
43. Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, C., Sun, M.: Graph neural networks: A review of methods and applications. AI open **1**, 57–81 (2020)