

# Tackling the Challenges in Scene Graph Generation With Local-to-Global Interactions

Sangmin Woo<sup>1</sup>, *Student Member, IEEE*, Junhyug Noh<sup>2</sup>, *Member, IEEE*, and Kangil Kim<sup>1</sup>, *Member, IEEE*

**Abstract**—In this work, we seek new insights into the underlying challenges of the scene graph generation (SGG) task. Quantitative and qualitative analysis of the visual genome (VG) dataset implies: 1) ambiguity: even if interobject relationship contains the same object (or predicate), they may not be visually or semantically similar; 2) asymmetry: despite the nature of the relationship that embodied the direction, it was not well addressed in previous studies; and 3) higher-order contexts: leveraging the identities of certain graph elements can help generate accurate scene graphs. Motivated by the analysis, we design a novel SGG framework, Local-to-global interaction networks (LOGINs). Locally, interactions extract the essence between three instances of subject, object, and background, while baking direction awareness into the network by explicitly constraining the input order of subject and object. Globally, interactions encode the contexts between every graph component (i.e., nodes and edges). Finally, Attract and Repel loss is utilized to fine-tune the distribution of predicate embeddings. By design, our framework enables predicting the scene graph in a bottom-up manner, leveraging the possible complementarity. To quantify how much LOGIN is aware of relational direction, a new diagnostic task called Bidirectional Relationship Classification (BRC) is also proposed. Experimental results demonstrate that LOGIN can successfully distinguish relational direction than existing methods (in BRC task), while showing state-of-the-art results on the VG benchmark (in SGG task).

**Index Terms**—Bidirectional relationship classification (BRC), scene graph generation, scene understanding, visual relationship detection.

## I. INTRODUCTION

**T**O UNDERSTAND a scene, inferring underlying properties such as the relationship between entities (in this work, we use the term “entity” to describe individual detected object instances to distinguish them from “object” in the semantic sense) is just as important as observing explicit

information about what and where entities are. However, most state-of-the-art visual recognition models focus on detecting individual entities in isolation [1]–[5], and they are still far from reaching the goal of capturing their relationships. In an effort to incorporate the relational reasoning ability into the model, a scene graph representation—a structured description that captures semantic summaries of entities and their relationships—has been presented recently [6]. Since then, a number of works have proposed deep network-based approaches for generating the scene graphs, confirming its importance to the field [7]–[13]. While scene graph representation holds tremendous promise, extracting scene graphs from images is known to be challenging.

In Section III, we first explore what the fundamental challenges of the task are as follows.

- 1) *Ambiguity*: We postulate the main cause of ambiguity is due to high intra- and low interclass variability of predicates. Although there is little visual difference between the images, the predicates can be different, and vice versa [see Fig. 1(a)]. Therefore, the model should be aware of the inconsistency between the visual and the actual predicate. In other words, we require a model that can recognize a subtle visual difference to differentiate the predicates and learn that the predicates can be the same in a completely different visual context.
- 2) *Asymmetry*: By nature, a relationship has a direction. Also, we can always define relationships in both directions. Nevertheless, we see that understanding the relational direction has not been well established in previous studies, and there is a lack of consideration on how to effectively address it. In this work, we are particularly interested in bidirectional relationships (BRs) with asymmetry [see Fig. 1(b)]. To analyze how much the model understands relational direction, we introduce a new diagnostic task called *BR classification (BRC)*.
- 3) *Higher-order contexts*: Often relations need to be considered with the contextual dependency of the whole scene beyond being defined as a pair-wise relation. Suppose there is a horse close to a man [see Fig. 1(c)]. Without any other clue, one might say that their relationship is “next to.” However, if the presence of other entities and relationships (e.g., hay, eating) is known, the relationship between the horse and the man is more likely to be “feeding.” To examine the benefits of higher-order contexts, we quantitatively analyze the amount of information gain given each graph component.

In Section IV, with the aforementioned issues in mind, we present a novel framework, **Local-to-Global Interaction**

Manuscript received 17 June 2021; revised 11 January 2022; accepted 12 March 2022. Date of publication 31 March 2022; date of current version 1 December 2023. This work was supported in part by the National Research Foundation of Korea (NRF) Grant by the Korean Government through MSIT under Grant 2019R1A2C1091077 and in part by the Institute of Information and Communications Technology Planning and Evaluation (IITP) Grant by the Korean Government through MSIT [Artificial Intelligence Graduate School Program, Gwangju Institute of Science and Technology (GIST)] under Grant 2019-0-01842. (Corresponding author: Kangil Kim.)

Sangmin Woo was with the Gwangju Institute of Science and Technology, Gwangju 61005, South Korea. He is now with the School of Electrical Engineering, Korea Advanced Institute of Science and Technology, Daejeon 34141, South Korea (e-mail: smwoo95@kaist.ac.kr).

Junhyug Noh is with the Computational Engineering Division, Lawrence Livermore National Laboratory, Livermore, CA 94550 USA (e-mail: noh1@llnl.gov).

Kangil Kim is with the School of Electrical Engineering and Computer Science, and the Artificial Intelligence Graduate School, Gwangju Institute of Science and Technology, Gwangju 61005, South Korea (e-mail: kangil.kim.01@gmail.com).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TNNLS.2022.3159990>.

Digital Object Identifier 10.1109/TNNLS.2022.3159990

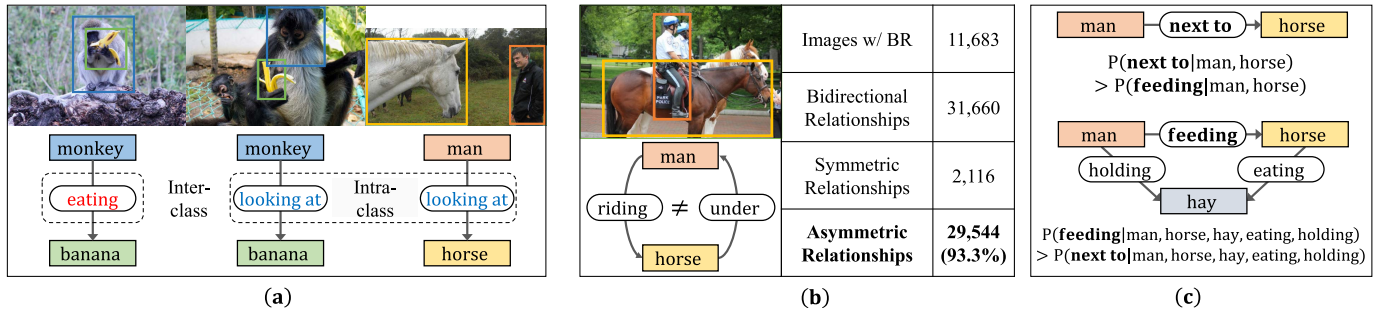


Fig. 1. Challenges in scene graph generation. (a) **Ambiguity**: different predicates may be visually similar (first two), and the same predicate may not be visually similar (last two). (b) **Asymmetry**: relationships have direction, and those relationships in the opposite direction are mostly different (asymmetric). (c) **Higher-order contexts**: the other components in the scene serve as contexts while predicting the relationship. Data statistics are based on the VG dataset [14].

**Networks (LOGINs)**. First, LOGIN highlights informative representation between three entity-level instances by weighing how much each pair-wise interaction contributes to relational representation. Second, direction awareness is baked into the model by fusing feature instances in a constrained order (e.g., subject precedes object). Third, LOGIN considers interaction between every scene graph element. The informative contexts essential to accurately predict each graph component are propagated to every graph component. Last but not least, we introduce Attract and Repel Loss, which effectively scales the variability within and between classes making the model robust against ambiguity. We explain this in more detail in Section IV-D. By design, LOGIN effectively leverages the complementarity of entity-level interactions and graph-level interactions.

Finally, in Section V, we evaluate our final model on both the visual genome (VG) benchmark and the BRC task. By ablating each network design, we observe that all design principles cooperate in generating visually grounded scene graphs. Unifying all design principles into a single framework, LOGIN achieved state-of-the-art results on the VG benchmark while outperforming competing approaches by a comfortable margin on the BRC task.

Our contributions can be summarized as follows.

- 1) Through quantitative and qualitative analysis on the VG dataset, we identify fundamental challenges in the SGG task: 1) ambiguity; 2) asymmetry; and 3) higher-order contexts.
- 2) We design a novel framework, LOGINs, to address the aforementioned issues, which achieved competitive results against state-of-the-arts on the VG benchmark.
- 3) To quantify and concretely see how well the model understands the relational direction, we introduce a new BRC task. Here, LOGIN significantly outperformed state of the art by a 6% of mean performance gain.

## II. RELATED WORK

Numerous works have actively studied the task of recognizing entities and their relationships in various forms. This includes entity localization from natural language expressions [15], human-entity interactions [16]–[18], or the more general tasks of visual relationship detection [19]–[26], and scene graph generation [27]–[40].

Among them, the scene graph generation has recently drawn much attention. The challenging and open-ended nature of the task lends itself to a variety of diverse methods. For example, refining entity and predicate labels using iterative message passing [27]; staging the generation process in three-step based on the observation that entity labels are highly predictive of predicate labels [28]; explicitly modeling interdependency among entire entities using bi-linear pooling [29]; leveraging the idea of proposal network [41] and graph convolution [42] jointly [30]; combining both visual and linguistic features to exploit linguistic analogies [32]; using statistical correlations between entity pairs and their relationships to regularize semantic space [33]; presenting a multiagent policy gradient method to replace standard cross-entropy loss and maximize a graph-level metric [35]; disentangle entity and predicate recognition, enabling subquadratic performance [37].

We shed light on three underlying challenges that have not been dealt with in-depth in previous studies: 1) ambiguity; 2) asymmetry; and 3) higher-order contexts (see Fig. 1). Similar to ours, the ambiguity issue has been addressed in [36], which is a proximal relationship ambiguity arising from multiple subject-object pairs being gathered nearby. On the other hand, we interpret it differently as visual and semantic ambiguity caused by high intraclass variance and low interclass variance. To the best of our knowledge, the asymmetry issue in SGG is explicitly and importantly addressed for the first time in this work. We believe that some works [20], [28], [34], [36], [38], [43] can also cope with relational directions, albeit they do not suggest an effective method. Higher-order context problems have been addressed a lot in previous SGG studies [27]–[30], but most focus on context utilization aspects. We rather take a slightly different approach to find the answer to the question, “Are we fully exploiting all the information available?” To this end, we examine the predictability of identity according to given graph components and then apply the most promising way we find to the context propagation step. In summary, we design LOGIN, an integrated framework based on the analysis, to tackle the challenges simultaneously.

## III. IDENTIFYING CHALLENGES IN SCENE GRAPH GENERATION

This section seeks quantitative insights on the underlying challenges of the SGG task by analyzing the VG dataset.

In particular: 1) **ambiguity** (see Section III-A): how intra- and interclass variance hinder clearly differentiating the predicate class boundary; 2) **asymmetry** (see Section III-B): how has relational direction been overlooked, and how can direction awareness be quantified; and 3) **higher-order contexts** (see Section III-C): what higher-order context should be considered to predict the identity of each graph element. Motivated by our findings, we design the LOGIN to better integrate local and global contexts, which will be described in more detail in Section IV.

#### A. Ambiguity

To gain insight into the VG scene graphs, we first examine the intraclass variance and interclass distance within and in between predicate categories. Specifically, we take a close look at label statistics (e.g., subject–object–predicate co-occurrence). Intraclass variance within  $i$ th predicate can be calculated as

$$\text{var}_{\text{intra}}(i) = \frac{1}{N^2} \sum_{k=1}^{N^2} (f_{ik} - \mu_i)^2. \quad (1)$$

The interclass distance between  $i$ th and  $j$ th predicate is normalized by the co-occurrence frequency

$$\text{dist}_{\text{inter}}(i, j) = \frac{\sum_{i=1}^M \sum_{j=1}^M \sum_{k=1}^{N^2} |f_{ik} - f_{jk}|}{\sqrt{\sum_{i=1}^M f_i} \sqrt{\sum_{j=1}^M f_j}} \quad (2)$$

where  $N$  and  $M$  are the number of entities and predicates, respectively.  $f_{ik}$  denotes the co-occurrence frequency of  $i$ th predicate and  $k$ th entity pairing, and  $\mu_i$  denotes the mean value.

The results are depicted in Fig. 2. From the figure, it can be observed that frequently occurring predicates tend to have high intraclass variance, which implies the dominant predicates can be used in various contexts repeatedly (i.e., even the same predicate can pair with various entity pair candidates). In contrast, most of the predicate–predicate pairs have similar subject–object co-occurrence distribution. In this case, even if subject–object identity is known, it becomes difficult to predict the predicate (i.e., predicates in different categories can often pair with the same entity pair).

In summary, even the same predicates may not be similar visually (see Fig. 1) nor semantically. Accordingly, solving these ambiguity issues could play a key role in generating accurate scene graphs.

#### B. Asymmetry

1) *Bidirectional Relationships*: Given two different entities  $A$  and  $B$ , if the both  $A \xrightarrow{\alpha} B$  and  $B \xrightarrow{\beta} A$  relationships are defined, we consider those relationships as *BRs*. Among them, if  $\alpha \neq \beta$ , we denote as *asymmetric relationships*, otherwise as *symmetric relationships*. Examples of BRs are shown in Fig. 3.

Among the total 108073 images in the VG Dataset [14], 11683 images contain 31660 BRs, which can be broken down into 29544 asymmetric relationships and 2116 symmetric

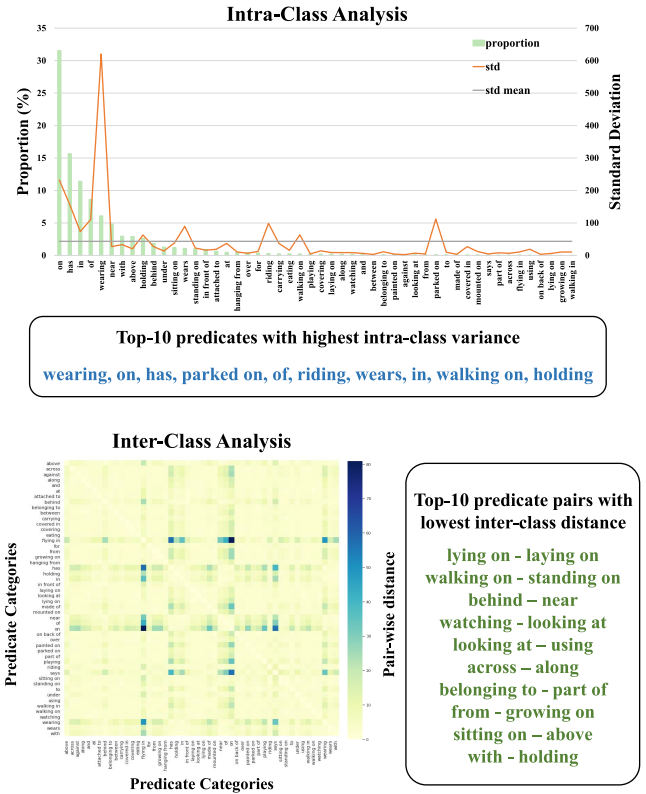


Fig. 2. Intra- and interclass analysis on predicates. (Top) Predicate labels are arranged in the order of proportion (green bar). More frequent predicates tend to have higher intraclass variance (orange line). (Bottom) Each block represents the degree to which predicate-predicate pair share the same entity pairs in color—the more overlapping entity pairs, the brighter the block color. Except for a few predicate pairs (e.g., on-flying in), most predicate pairs have low interclass distance—the closer the brighter. Axis labels best viewed zoomed in on screen.

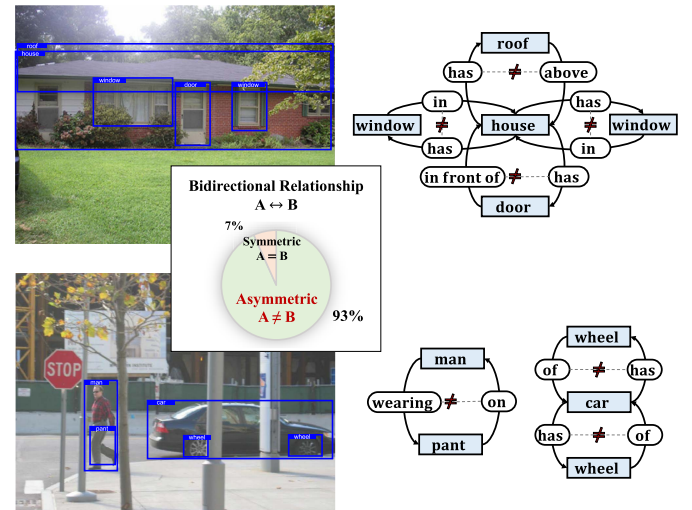


Fig. 3. Examples of BRs in VG dataset. (Left) Input images with ground-truth bounding boxes (right) corresponding BR scene graph. Nodes and edges are colored in blue and green, respectively. As can be seen in the figure, most BRs are asymmetric.

relationships [see Fig. 1(b)]. Since the majority of relationships are asymmetric ( $\sim 93.3\%$ ), modeling the relational direction (with regard to the entity orderings) is crucial (see Fig. 7).



2) *Modeling Relational Direction*: One straightforward approach to obtain a visual representation of a predicate is to use a union appearance feature<sup>1</sup> directly, which is the form used by many previous works [19], [24], [27], [30]–[35], [44]. Using only the union feature is straightforward and reflects the holistic representation, but it entails a fatal problem. For example, even the position of two entities is reversed, the union feature remains the same thus cannot embody directionality without the assistance of external features (e.g., spatial coordinates, contexts). This weakness is more pronounced when predicting relationships in opposite directions at the same time.

3) *Diagnosis of Direction Awareness*: It is common sense that all relationships have directions and can always be defined in both directions. However, since most of the relationships that make up the VG dataset are uni-directional, a rigorous analysis on the direction awareness of model is limited. In other words, good performance can be achieved in the VG dataset without due consideration of the direction awareness.

To this end, we introduce a new diagnostic task called *BRC* to quantify and concretely see how well the model understands the relational direction. The task is solely based on the collected images containing BRs. Therefore, in all cases, good performance can only be achieved by understanding the direction.

What we want to observe in the BRC benchmark is how much the model understands the direction of the relationship. Among the three common criteria for evaluating performance in SGG, *SGGen* includes not only predicate prediction but also object localization and object class prediction, making it difficult to evaluate direction prediction intensively. Likewise, since *scene graph classification (SGCls)* includes class prediction of objects, it may be difficult to strictly verify the directional understanding. Therefore, we adopt *predicate classification (PredCls)* evaluation criterion that measures only predicate predictions.

### C. Higher-Order Contexts

To investigate the benefits of higher-order context, we measure how much information is gained given the identity of different scene graph elements. Motivated by [28], we plot the likelihood of guessing labels of target element given labels of other graph elements in Fig. 4. In addition to node-conditioned guessing performed on prior work [28], we further analyze the predictability improvement given the edge identity. To disentangle the significance of semantic knowledge from image cues (e.g., appearance, spatial), no image features are used and are guessed using only label statistics (i.e., subject-object-predicate co-occurrence). A higher curve implies that given graph elements are more decisive in guessing the target element.

In the case of edge, it is greatly affected by the identity of neighboring nodes, consistent with our intuition. What is more noteworthy here is that even only one edge in the opposite direction is known, nearly 90% accuracy can be achieved within just five guesses. It can also provide complementary

<sup>1</sup>A union appearance feature is pooled from the Region of Interest (RoI) feature that tightly encompasses two (subject and object) entities.

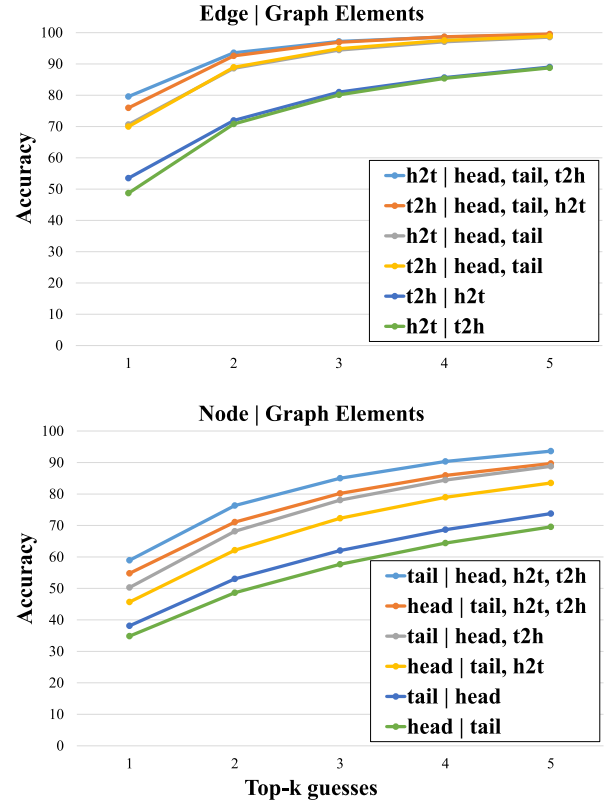


Fig. 4. How much information does each graph component contain? The figures show the likelihood of guessing the label of target element given the identity of neighboring graph components—head, tail, and edge. Guesses were made by looking up the empirical distribution over label statistics in the training set (e.g., top- $k$  frequent classes given graph elements). *h2t* refers to the edge from the head node to the tail node, and *t2h* refers to the edge of the opposite direction.

information in determining the identity of the target edge when given with the neighboring nodes' identity.

In the case of node, it has less correlation with adjacent graph elements than edges. However, as shown in the figure, a significant amount of information can be obtained whenever the identity of adjacent graph elements is known one by one. This fact motivates the use of as much information as possible to correctly recognize the identity of each element.

To sum up, we see that both node and edge can most effectively exploit inductive bias when utilizing all the identities of adjacent graph elements.

## IV. LOCAL-TO-GLOBAL INTERACTION NETWORK

Based on the analysis in Section III, we design a novel framework LOGIN that aims to handle said issues in a bottom-up manner. Each building block in LOGIN is specialized in tackling specific challenges and also works complementary to each other. An overview of LOGIN is shown in Fig. 5.

*Problem Setup*: Given an image  $I$ , the detector predicts a set of entity proposals. For each entity proposal, it outputs an RoI Aligned [2] visual appearance feature  $a_i \in \mathbb{R}^{256 \times 7 \times 7}$ , a bounding box prediction  $b_i \in \mathbb{R}^4$ , and initial classification logit  $c_i \in \mathbb{R}^{151}$ . In practice, a standard entity detector Faster R-CNN [41] is used as a bounding box model.

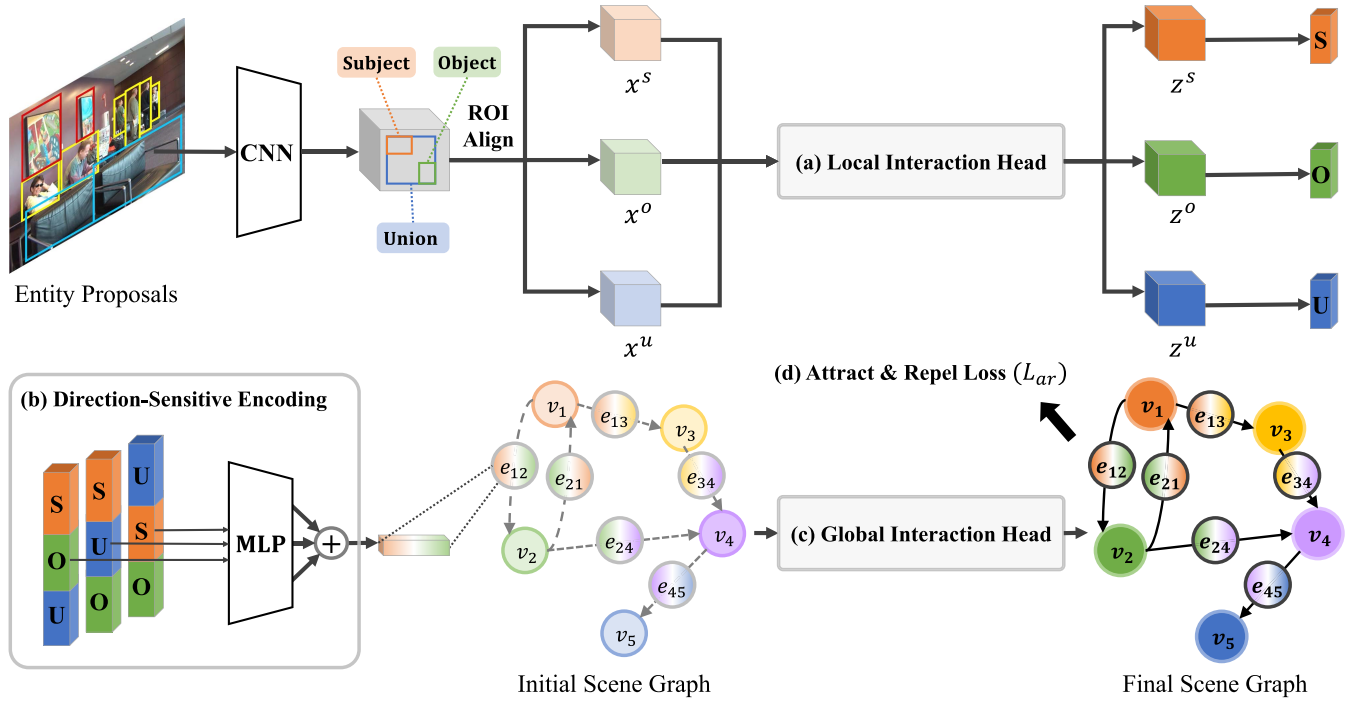


Fig. 5. High-level overview of LOGIN. (a) **Local interaction head** (see Section IV-A): locally, interactions extract the essence between three instances—subject, object, and background. (b) **Direction sensitive encoding (DSE)** (see Section IV-B): model become aware of relational direction by constraining the input order. (c) **Global interaction head** (see Section IV-C): globally, interactions encode the contexts between every graph components—nodes and edges, allowing the model to encode richer contextual information. (d) **Attract and Repel Loss** (see Section IV-D): embeddings of each predicate categories are gathered into compact and well separated clusters by the loss. Combining all together, we build an end-to-end, unified framework that predicts a visually grounded scene graph.

Starting from a set of entity proposals (equivalent to a set of nodes in scene graphs), visual features are pooled from the subject and object boxes that form a relationship and from a union box to utilize contextual information (e.g., background) via RoI-Align operation, then predict the node and edge labels through scene graph generation head in turn.

The initial scene graph comprises a set of node representations  $\mathcal{N}$  and a set of edge representations  $\mathcal{E}$ . The  $i$ th initial node representation  $x_i \in \mathcal{N}$  is obtained by fusing three important cues in the image:  $x_i = \phi([a_i || b_i || c_i])$ , where  $\phi$  is an embedding function and  $||$  denotes concatenation operation. The edge representations  $\mathcal{E}$  are obtained through several stages of process that will be described in the following.

The final scene graph is composed of a set of node label distributions  $\mathfrak{N} \in \mathbb{R}^{N \times 151}$  (including *no-object* class) and a set of edge label distributions  $\mathfrak{E} \in \mathbb{R}^{M \times 51}$  (including *no-relation* class), where  $N$  and  $M$  is the number of total nodes and edges, respectively.

#### A. Local Interaction Head

We posit the underlying vulnerability to ambiguity stems from the inability to capture subtle yet discriminative representations. Inspired by the recent successes of attention-based fine-grained recognition works [45]–[51], where the intraclass variance is usually high and vice versa for interclass, we adopt the idea of attention mechanism. In particular, as for local-interaction head (LIH), we formulate the instance-level interaction as a non-local operation [48]. LIH learns to highlight relationship-centric representations and suppress the

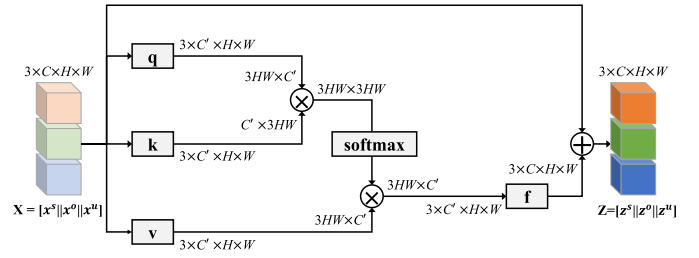


Fig. 6. LIH learns what ( $3 \times C$ ) and where ( $H \times W$ ) to attend. It adaptively learns to emphasize informative representation between three entity-level instances by weighing how much each pair-wise interaction contributes to relational representation.

noise since the non-local operation considers all individuals to compute responses to the target individual.

The intensity of pair-wise interaction is calculated over the three entity (node) features  $\{x^s, x^o, x^u\}$  (each refers to subject, object, and union, see Fig. 6). Given concatenated features  $\mathbf{X} = [x^s || x^o || x^u]$ , LIH outputs refined features  $\mathbf{Z} = [z^s || z^o || z^u]$ . The interaction intensity between  $i$  and  $j$ th individual is computed by the embedded Gaussian kernel ( $e^{\mathbf{q}(\cdot)}$ ,  $e^{\mathbf{k}(\cdot)}$ ) and normalized by the sum

$$\alpha_{ij} = \frac{e^{\mathbf{q}(x_i)^T \mathbf{k}(x_j)}}{\sum_{\forall j} e^{\mathbf{q}(x_i)^T \mathbf{k}(x_j)}}. \quad (3)$$

The interaction intensity  $\alpha_{ij}$  is multiplied with the representation of  $i$ th individual  $\mathbf{v}(x_i)$  followed by a transformation

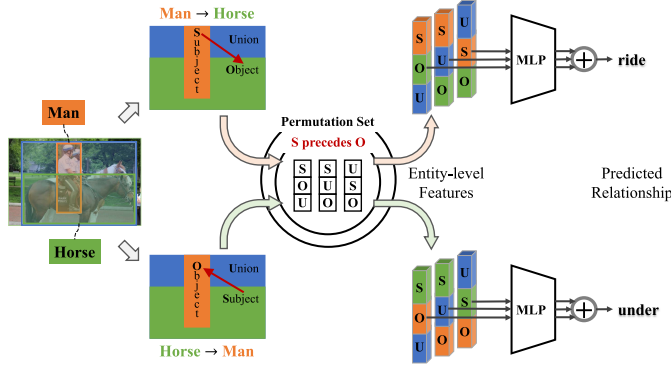


Fig. 7. Direction sensitive encoding. Between a pair of entities, the subject and the object can be switched, and two opposite-direction relationships are usually asymmetric. For example, the relationships of  $\text{Man} \rightarrow \text{Horse}$  and  $\text{Horse} \rightarrow \text{Man}$  are generally different. For both-side relationships, we fed all possible *permutations* of entity-level features (e.g., subject, object, and union) that satisfy the conditions of “subject precedes object” into the same MLP. Note that although both relationships follow the same condition, color combinations (e.g., orange, green, and blue) vary with direction. The final relationship is predicted after summing all the outputs of the MLP. During training, the MLP learns to generate different outputs for the opposite-direction relationships, thus it becomes aware of the relational direction.

function  $\mathbf{f}(\cdot)$ . The output of LIH operation is given by

$$z_{ij} = \mathbf{f}(\alpha_{ij}^T \mathbf{v}(x_i)) + x_i. \quad (4)$$

For the sake of better gradient flow while learning the LIH, a residual-connection ( $+x_i$ ) is added. In practice,  $1 \times 1 \times 1$  convolutional operations is used for all embedding functions ( $\mathbf{q}(\cdot)$ ,  $\mathbf{k}(\cdot)$ ,  $\mathbf{v}(\cdot)$ ,  $\mathbf{f}(\cdot)$ ).

### B. Encoding Direction-Awareness

Before moving on to the next stage, there is an open choice on how to fuse three instance features  $\{z^s, z^o, z^u\}$  obtained earlier to initialize a graph-level predicate (edge) representation. As suggested in [52], summing up all possible permutations of instance features could be a generic method for relational inference. The effectiveness of using permutation has been empirically demonstrated in prior works [53]–[55], but the directionality cannot be guaranteed since summing is commutative. In other words, the permutation sets are identical even if the ordering of two instances are reversed (i.e., the identity of subject and object are switched). A simple sidestep is to use the embeddings of concatenated instance features, which is the form used in the several previous works [20], [28], [34], [36], [43]. However, this also has the disadvantage of losing the benefits of permutation.

We would like the LOGIN to be *equivariant* under the subject–object ordering (i.e., relational direction) while being *invariant* to the permutation. Let the interaction between two entities as a set of permutations,  $S$ , and directional relationship as any subset except the empty and the universal set,  $r_i \subset S$ , where  $r_i \neq \phi$  and  $r_i \neq S$ ,  $i \in \{f, b\}$ . If the two opposite-sided relationship (forward and backward) subsets are disjoint,  $r_f \cap r_b = \phi$ , and their union is universal,  $r_f \cup r_b = S$ , the relationship encoding of two subsets can always be semantically distinguished. Under these premises, we specifically use a regulated set of permutations in which

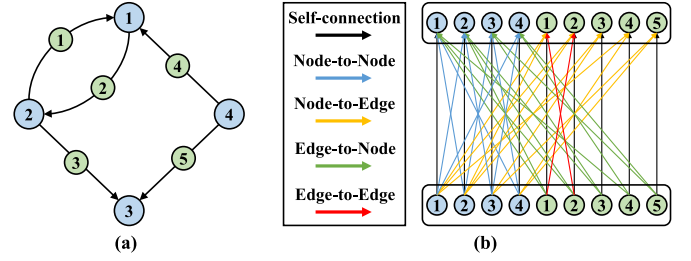


Fig. 8. Illustration of layer-wise propagation of global-interaction head. Nodes and edges are colored in blue and green, respectively. The layer-wise context propagation of GIH in (a) scene graph can be represented as a bipartite graph in (b). As a comparison, conventional GCN [42] and GAT [56] only consider node-wise propagation (black and blue edges) and are unable to leverage edge information. An experimental comparison of GIH with GCN and GAT is in Table VI.

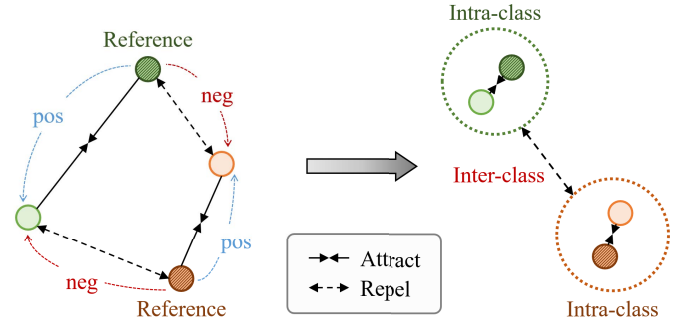


Fig. 9. Mechanism of Attract and Repel loss. The reference embeddings attract the positives and repel the negatives, making intraclass distribution dense and interclass distribution sparse. For simplicity, we visualize only one positive and negative instance.

the subject always precedes the object— $\{\text{SOU}, \text{SUO}, \text{USO}\}$ —to represent a relationship in one direction. This simple strategy guarantees that subject only appears in the first two bins, and that object only appears in the last two bins. Thus, the model can clearly distinguish between forward and backward relationships while sharing entity instance features. Note that this strategy is just a straightforward method to make half of the entire permutation set represent the forward direction and the other half represent the backward direction, and it does not matter which combination of permutations is used.

Formally, the three instances from an input set  $\{z^s, z^o, z^u\}$  are concatenated in a constrained order, providing inherent bias of the directionality. They are then transformed via shared multi-layer perceptron (MLP) (denoted as  $\varphi$  in the below equation) and additively fused to make the predicate representation invariant to the input permutations.  $i$ th predicate representation  $e_i \in \mathcal{E}$  can be obtained as

$$e_i = \sum_{j,k,l \in \{z^s, z^o, z^u\}} \varphi([j \parallel k \parallel l]) \quad (5)$$

where  $z^s$  precedes  $z^o$  and  $j \neq k \neq l$ .

### C. Global Interaction Head

From a graph perspective, the fully connected layer can be seen as the most basic form of message passing network with all nodes connected, but it is known to be not effective in learning the graph. For effective context aggregation, well structuring the message paths (i.e., connectivity

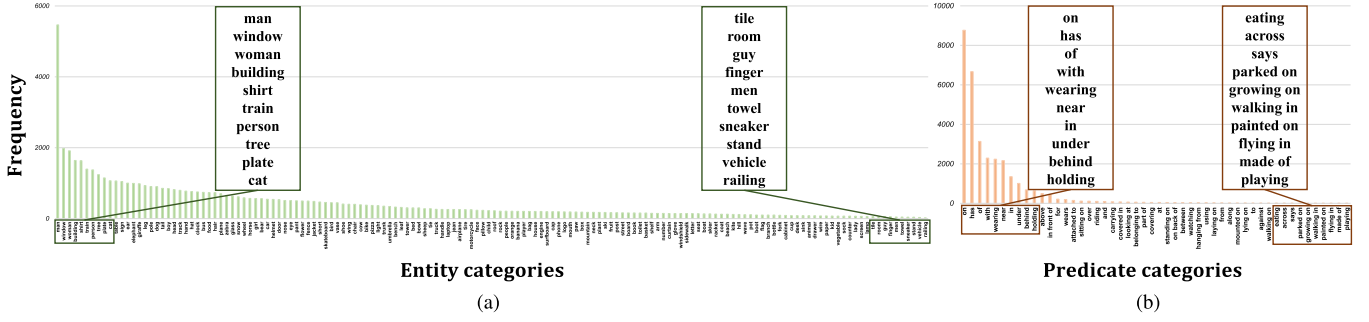


Fig. 10. Distribution of categories in the BR dataset. (a) Frequency of entity categories and (b) predicate categories. For both entity and predicate, the top-ten categories and the bottom-ten categories are highlighted based on frequency. Axis labels best viewed zoomed in on screen.

between nodes) is the key issue. Based on the observation in Section III-C, we design a Global Interaction Head (GIH) that enables effective message flow between informative graph components. We formulate the graph-level interaction with global message passing scheme [42], [56], [57].

To maintain a structured representation of a scene graph, we utilize local connectivity information in the form of a block matrix with four quadrants  $A \in \mathbb{R}^{(N+M) \times (N+M)}$ . Each quadrant from top left to bottom right indicates whether the *node-node* ( $A_{n-n} \in \mathbb{R}^{N \times N}$ ), *node-edge* ( $A_{n-e} \in \mathbb{R}^{N \times M}$ ), *edge-node* ( $A_{e-n} \in \mathbb{R}^{M \times N}$ ), and *edge-edge* ( $A_{e-e} \in \mathbb{R}^{M \times M}$ ) are connected (1) or not (0)—the number of nodes and edges are denoted as  $N$  and  $M$ , respectively. We consider that all node pairs and node-edge pairs that make up a relationship are interconnected. In the case of *edge-edge*, they are considered to be connected when the opposite direction edge exists, although there is no explicit connection on the graph (see Fig. 8)

$$A = \begin{bmatrix} A_{n-n} & A_{n-e} \\ A_{e-n} & A_{e-e} \end{bmatrix}. \quad (6)$$

To preserve the original message, identity matrix (self-connection) is added to  $A$ , resulting  $\hat{A} = A + I$ . An initial graph-level feature matrix  $\mathcal{G}^{(0)} \in \mathbb{R}^{(N+M) \times D}$  is defined as

$$\mathcal{G}^{(0)} = \begin{bmatrix} \mathcal{N} \\ \mathcal{E} \end{bmatrix}. \quad (7)$$

The  $l$ th layer-wise propagation rule for GIH is defined as

$$\mathcal{G}^{(l)} = \begin{cases} \max(0, \tilde{A}\mathcal{G}^{(l-1)}W^{(l-1)}), & l = \text{odd} \\ \mathcal{G}^{(l-2)} + \max(0, \tilde{A}\mathcal{G}^{(l-1)}W^{(l-1)}), & l = \text{even}. \end{cases} \quad (8)$$

We add residual connections between the layers for a better optimization [58]. Multi-layer GIH can perform long-range multi-hop communication, effectively modeling the desired higher-order relational reasoning. While training, weight matrix  $W \in \mathbb{R}^{D \times D}$  is learned by gradient.

Finally, the upper  $N$  rows ( $\mathcal{N}' \in \mathbb{R}^{N \times D}$ ) and the lower  $M$  rows ( $\mathcal{E}' \in \mathbb{R}^{M \times D}$ ) of the output matrix are softmaxed and used to predict entity and predicate labels.

#### D. Attract and Repel Loss

We introduce an Attract and Repel loss to explicitly handle the intra- and interclass variance. The conceptual mechanism of Attract and Repel loss is shown in Fig. 9. In a nutshell, if the identities of the input and reference embeddings are

the same (i.e., category matches), the loss forces them to *attract* each other; otherwise, the loss compels them to *repel* each other. The reference embeddings can be divided into two types: we refer to the running mean of the matched reference embeddings as positive (pos), and negative (neg) for that of non-matched. Note that the reference type is only an abstract distinction and can be vary depending on the identity of the input embedding. As the input embeddings are learned to approach the positive and move farther away from the negatives, the distribution within categories becomes dense, and between categories becomes sparse. As a result, the loss gathers the embeddings of each class into compact and well-separated clusters. Since most BRs are asymmetric (i.e., identities of predicates in opposite directions are mostly different) as we have seen in Section III-B, the loss has the potential benefit in predicting predicates in opposite directions differently. Formally, at  $t$ th batch, given a set of predicate embeddings  $\mathcal{E}^{(t)} = \{e_1^{(t)}, \dots, e_M^{(t)}\}$ , a set of references  $\mathcal{R}^{(t)} = \{r_1^{(t)}, \dots, r_{S1}^{(t)}\}$  is adjusted with the following update rule:

$$r_m^{(t)} = \frac{r_m^{(t-1)} * \mathbb{N}(r_m^{(t-1)}) + \sum_{i \in \text{pos}} e_i^{(t)} - \sum_{j \in \text{neg}} e_j^{(t)}}{\mathbb{N}(r_m^{(t-1)}) + \mathbb{N}(\text{pos}) + \mathbb{N}(\text{neg})} \quad (9)$$

where  $\mathbb{N}(\cdot)$  denotes the number of embeddings considered in reference update. Finally, the input predicate embeddings are adjusted with the following Attract and Repel loss:

$$\mathcal{L}_{\text{ar}} = \sum_m \sum_{i \in \text{pos}} \left( 1 - \frac{r_m^{(t)} \cdot e_i^{(t)}}{|r_m^{(t)}| |e_i^{(t)}|} \right) + \sum_m \sum_{j \in \text{neg}} \left( \frac{r_m^{(t)} \cdot e_j^{(t)}}{|r_m^{(t)}| |e_j^{(t)}|} \right). \quad (10)$$

#### E. Loss Function

LOGIN can be trained in an end-to-end manner, allowing the network to predict bounding boxes, entity categories, and relationship categories at once. The total loss function for an image is defined as

$$\mathcal{L}_{\text{image}} = \mathcal{L}_{\text{ent}} + \mathcal{L}_{\text{pred}} + \mathcal{L}_{\text{ar}} \quad (11)$$

where  $\mathcal{L}_{\text{ent}}$  and  $\mathcal{L}_{\text{pred}}$  are both cross-entropy loss for entity and predicate classification, respectively.  $\mathcal{L}_{\text{ar}}$  stands for the Attract and Repel loss. By default, hyperparameters of joint loss function are set as 1:1:1.



## V. EXPERIMENTS

In this section, we conduct comprehensive studies to validate the efficacy of LOGIN. We perform extensive ablation experiments to thoroughly demonstrate the effectiveness of each building block of LOGIN. LOGIN is evaluated on VG [14] benchmark and achieves state-of-the-art results. Notably, in our proposed BRC task, LOGIN successfully distinguishes asymmetric relationships and is more accurate than existing methods.

The model referred to as the BASELINE in this section is a model without any proposed design principles. It directly predicts the entity and predicate categories from the RoI-aligned visual features of entity instances and that of union of two entity instances, respectively.

### A. Settings

1) *Model Parameter and Training Details*: For a fair comparison, most of the settings and details follow pioneer work [27], [28]. We adopt the Faster R-CNN [41] detector with VGG backbone [5]. Following [28], we use per-class non-maximum suppression (NMS) to reduce the number of entity proposals. The number of entity proposals is 64 (i.e.,  $N = 64$ ). We optimize the model using stochastic gradient descent (SGD) with the following details: initial learning rate ( $1e-3$ ), momentum (0.9), and weight decay ( $5e-4$ ). We first pre-train the detector on VG Dataset and then train the proposed scene graph generation head while fixing the detector weight. To model geometric relationships, we first concatenate two extra channels with coordinates hard-coded ( $2 \times 7 \times 7$ ) to the initial visual representation and then pass them through a convolutional layer [59]. As for the Attract and Repel loss, we sample negatives by the number of positives to avoid being heavily affected by negatives.

2) *VG Dataset*: We train and evaluate LOGIN on VG Dataset [14]. We use the publicly released pre-processed data (train and test split is 75 and 32 K) [27]. The number of entity and predicate categories are 150 and 50, respectively.

3) *BR Dataset*: We build a BR dataset to evaluate the direction awareness of the model. The BR dataset is a subset of VG dataset and is created by filtering out relationships with only one edge between the two nodes. As a result, the BR dataset always includes the relationships that have two bidirectional edges between the nodes (e.g., man  $\xrightarrow{\text{riding}}$  horse, horse  $\xrightarrow{\text{under}}$  man). As shown in Fig. 1, about 93% of bidirectional edges form different relationships depending on the direction (i.e., direction-sensitive), and only about 7% of bidirectional edges have the same relationship regardless of direction (i.e., direction-agnostic). The distribution of BR Dataset is shown in Fig. 10. Here, the five most frequent entity categories and predicate categories are “man (5466), window (1976), woman (1912), building (1640), shirt (1632)”, and “on (8766), has (6669), of (3137), with (2292), wearing (2238),” respectively. Note that the top-five predicate categories account for about 73% of the total predicates. This shows that the dominant predicate categories are often used in various contexts repeatedly, implying that variance may be high even within the same predicate category. That is to say, this biasness supports our argument that dealing with *ambiguity* issue is essential.

4) *Evaluation Setup*: Model is evaluated with the following three standard evaluation criteria [27].

- 1) *PredCls*: Given ground truth boxes and labels, predict edge labels.
- 2) *SGCls*: Given ground truth boxes, predict box, and edge labels.
- 3) *Scene Graph Generation (SGGen)*: Predict boxes, box labels, and edge labels.

As for SGG, following the prior works [27], [28], [30], we use Recall@K (R@K) as an evaluation metric since mean average precision (mAP)-like metrics are not appropriate due to the sparse annotation in VG. Specifically, we use image-wise Recall@{20, 50, 100}, which computes the fraction of ground-truth triplets found in the top- $K$  predicted triplets. We also adopt the mean Recall@K (mR@K) metric [33], [34], [38] for evaluation, which retrieves each individual predicate and then averages R@K over all predicate categories.

As for BRC, conventional triplet recall-based metrics only consider uni-direction, making it difficult to make a rigorous evaluation of direction awareness. To this end, we have come to introduce a new metric called *pair-wise Recall (pR@K)* that fits the BRC task. The proposed metric is considered to be “matched” only when BRs are both correct. Formally, the pR@K calculates the fraction of the total amount of matched BRs

$$\text{pR@K} = \frac{|\{\text{top-K predicted BRs}\} \cap \{\text{total BRs}\}|}{|\{\text{total BRs}\}|}. \quad (12)$$

This constraint severely penalizes if the relationship predictions in the opposite direction are the same. Therefore, models without direction awareness cannot receive a high score on this metric. For example, if only union features are used, there is no chance that asymmetric relationships are correctly predicted since only the same results are output for BRs. To get a high score from this metric, the model needs direction awareness that is essential to correctly predict asymmetric relationships that account for most BRs in the BR datasets. Specifically, we use pR@{2, 4, 8, 16} in the BRC task since only a few BRs are annotated per image ( $\sim 3$  BRs/image).

### B. Comparison With State-of-the-Art

1) *Scene Graph Generation (SGG)*: The Recall performance of the proposed method and existing methods are compared in Table I for each evaluation criterion. We compare LOGIN with the recent approaches [27], [28], [30], [33]–[36]. While LOGIN appears to show competitive results against the state-of-the-arts in all criteria, note that there is no specific method that achieves the best performances in every evaluation criteria, making it difficult to judge the superiority among the SGG methods.

We also benchmark LOGIN under the mean Recall (mR@K) criteria. The results are shown in Table II. The mean Recall is measured by averaging the Recall per each class for the entire classes. Therefore, unlike conventional Recall (R@K), it is irrelevant to the number of samples in each class, and even if high performance is obtained in a class with a large number of samples, it is difficult to achieve good values if low performance is obtained in a class with a small number of samples. That is, every class should obtain



TABLE I  
COMPARISON WITH THE STATE-OF-THE ARTS ON VG BENCHMARK.  $R@k$  DENOTES RECALL IN THE TOP- $k$  PREDICTIONS

Models	<i>SGen</i>			<i>SGCls</i>			<i>PredCls</i>		
	R@20	R@50	R@100	R@20	R@50	R@100	R@20	R@50	R@100
IMP [27]	-	3.4	4.2	-	21.7	24.4	-	44.8	53.0
MOTIFNET [28]	21.4	27.2	30.3	32.9	35.8	36.5	58.5	65.2	67.1
GRAPH R-CNN [30]	-	11.4	13.7	-	29.6	31.6	-	54.2	59.1
KERN [33]	-	27.1	29.8	-	36.7	37.4	-	65.8	67.6
CMAT [35]	22.1	27.9	31.2	35.9	<b>39.0</b>	39.8	60.2	66.4	68.1
VCTREE [34]	22.0	27.9	31.3	35.2	38.1	38.8	60.1	66.4	68.1
RELDN [36]	21.1	<b>28.3</b>	<b>32.7</b>	<b>36.1</b>	36.8	36.8	<b>66.9</b>	<b>68.4</b>	68.4
<b>LOGIN (OURS)</b>	<b>22.2</b>	28.2	31.4	35.5	38.8	<b>40.5</b>	61.1	66.6	<b>68.7</b>

TABLE II  
SGG RESULTS ON MEAN RECALL (MR@K).  $mR@k$  DENOTES AVERAGE  $R@K$  OVER ALL PREDICATE CATEGORIES

Models	<i>SGen</i>			<i>SGCls</i>			<i>PredCls</i>		
	mR@20	mR@50	mR@100	mR@20	mR@50	mR@100	mR@20	mR@50	mR@100
IMP [27]	-	3.8	4.8	-	5.8	6.0	-	9.8	10.5
MOTIFNET [28]	4.2	5.7	6.6	6.3	7.7	8.2	10.8	14.0	15.3
KERN [33]	-	6.4	7.3	-	9.4	10.0	-	17.7	19.2
VCTREE [34]	5.2	6.9	8.0	8.2	10.1	10.8	14.0	17.9	19.4
<b>LOGIN (OURS)</b>	<b>5.9</b>	<b>7.7</b>	<b>9.1</b>	<b>8.6</b>	<b>11.2</b>	<b>12.4</b>	<b>16.0</b>	<b>19.2</b>	<b>22.3</b>

TABLE III  
COMPARISON WITH RECENT APPROACHES IN THE BRC TASK

Models	<i>PredCls</i>			
	pR@2	pR@4	pR@8	pR@16
IMP [27]	6.3	9.1	12.2	15.0
MOTIFNET [28]	7.7	11.5	15.9	19.5
GRAPH R-CNN [30]	7.9	11.7	16.3	21.0
KERN [33]	7.7	12.1	16.7	20.7
VCTREE [34]	8.0	11.9	16.1	21.0
RELDN [36]	8.0	12.5	16.4	20.8
<b>LOGIN (OURS)</b>	<b>8.6</b>	<b>13.1</b>	<b>17.6</b>	<b>21.1</b>

a good overall recall to achieve high performance. In short, it is important to accurately predict the class with few data, especially in tail, among long-tailed VG dataset. The long-tailed distribution of VG dataset also implies that the dominant predicates frequently appear in multiple contexts. Thus, it is also related to the ambiguity issue. We see that LOGIN consistently outperforms recent methods in mean Recall criteria (see Table VII for Recall of individual predicate), implying that our system effectively deals with ambiguity issue.

2) *BR Classification*: To independently evaluate the direction-awareness of the model, we specifically use *PredCls* criteria, which is orthogonal to the entity detection. We compare LOGIN with recent approaches [27], [28], [30], [33], [34], [36]. The results are summarized in Table III. Here,

although [27], [30], [33] use the initial predicate representation as a union feature, they enable understanding of relational direction by incorporating contexts with iterative bipartite message passing, attentional graph convolution, and knowledge embedded routing, respectively. By using direction sensitive embedding and contextual information at the same time, LOGIN can outperform the recent methods by a large margin (6% of mean performance gain compared to the state-of-the-art), implying that directional bias as well as contexts are crucial in recognizing direction. LOGIN is in a competitive position for VG dataset, which mainly contains uni-directional relationships, but significantly improves performance, especially for BRs, which are common in the real world.

### C. Quantitative Analysis

1) *Model Ablations*: We consider several ablations to investigate the importance of the major design choices in Table IV(a). For clarity, we show the performance in the SGG task and the BRC task in a single table. Exp 1 is the result of a vanilla version of LOGIN, i.e., BASELINE, which shows the abysmal result, especially in the BRC setting. This means that the BASELINE has no understanding of relational direction at all; thus, it can only predict symmetric relationships correctly. Exps 2–5 examine the individual contributions of each model component. Especially, LIH (Exp 2) and GIH (Exp 4) have a significant impact on both SGG and BRC settings. It is noteworthy that contextual information (driven from GIH) also plays a key role in recognizing directions. We can see in Exp 3 that DSE is relatively unremarkable in

TABLE IV  
(a) ABLATION STUDIES ON NETWORK DESIGN. (b) OPTIMAL VARIABLE SEARCH

(a)												(b)				
Exp	Ablations				SGCls			PredCls				Variables	SGCls			
	LIH	DSE	GIH	$\mathcal{L}_{ar}$	R@20	R@50	R@100	pR@2	pR@4	pR@8	pR@16		R@20	R@50	R@100	
1					30.8	34.7	36.2	0.2	0.4	0.7	1.3	Feature	AVGPOOL	34.3	38.4	40.2
2	✓				33.5	37.5	39.6	8.0	11.5	17.0	20.3		MAXPOOL	34.1	38.4	40.1
3		✓			31.8	36.2	37.7	8.1	11.5	16.4	20.1		<b>FLATTEN</b>	<b>34.5</b>	<b>38.8</b>	<b>40.5</b>
4			✓		33.2	38.4	39.9	8.2	12.0	16.9	20.6	GIH	2-LAYERS	34.1	38.4	39.9
5				✓	31.2	35.6	36.9	7.4	11.1	16.0	19.7		<b>4-LAYERS</b>	<b>34.5</b>	<b>38.8</b>	40.5
6	✓	✓	✓		34.4	38.5	40.3	8.4	12.9	17.5	21.0		6-LAYERS	34.1	38.5	<b>40.6</b>
7	✓	✓	✓	✓	<b>34.5</b>	<b>38.8</b>	<b>40.5</b>	<b>8.6</b>	<b>13.1</b>	<b>17.6</b>	<b>21.1</b>					

TABLE V  
COMPARISON OF FEATURE FUSION METHODS. (a) SGCs RESULTS ON VG DATASET. (b) BRC RESULTS ON BR DATASET

(a)					(b)				
Fusion	SGCls				Fusion	PredCls			
	R@20	R@50	R@100			pR@2	pR@4	pR@8	pR@16
BASLINE	30.8	34.7	36.2			0.2	0.4	0.7	1.3
w/o Permutation	33.6	38.0	39.9			8.2	12.2	17.0	20.6
Sequential	34.1	38.4	40.1			8.3	12.7	17.3	20.9
<b>Parallel (Ours)</b>	<b>34.5</b>	<b>38.8</b>	<b>40.5</b>		<b>Parallel (Ours)</b>	<b>8.6</b>	<b>13.1</b>	<b>17.6</b>	<b>21.1</b>

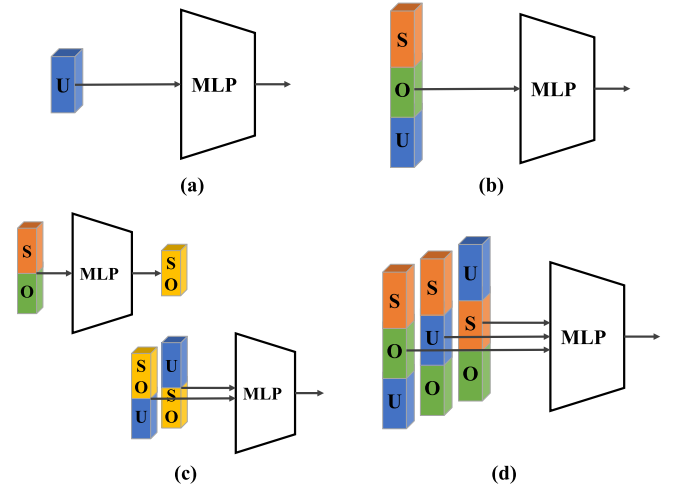


Fig. 11. Illustration of feature fusion methods to obtain initial predicate representation. (a) BASELINE: use union feature only. (b) W/o Permutation: concatenate all and fuse them without permutation. (c) Sequential: fuse subject and object first, and then with union. (d) Parallel: fuse all the permutations at once where the subject precedes object. Here, *S*, *O*, *U*, respectively, denotes subject, object, and union.

SGG settings, while it improves performance in BRC settings by a large margin. Although the unary effect of Attract and Repel Loss  $\mathcal{L}_{ar}$  (Exp 4) is not significant, using the loss with other components (Exp 7) can further push the performance than without it (Exp 6). When all model components are combined (Exp 7), the model achieves the best performance in both SGG and BRC tasks, which implies that each component contains an orthogonal factor that complementarily boosts the performance.

2) *Optimal Variables*: We conduct experiments to decide optimal variables of LOGIN in Table IV. The optimal feature extraction method is first investigated. Here, flattening the feature maintains richer information than pooling, thus shows the best results among the three choices: AVGPOOL, MAXPOOL, FLATTEN. Then we examine the optimal number of layers in GIH: four-layers produce the best results. Stacking multiple layers enables multi-hop communication, though it also increases the chance of introducing noisy information. On the other hand, stacking few layers cannot fully capture the higher-order contexts.

3) *Design Choices of DSE*: In this experiment, we further explore the four design choices of Direction-Sensitive Encoding. Specifically, we investigate two approaches that have been adopted in most existing SGG literature: 1) using only a union feature [19], [24], [27], [30]–[33], [35], [44]

(BASELINE); 2) fusing subject, object, and union without-permutations [20], [28], [34], [36], [38], [43]—and two variants of subject, object, and union fusion under the *subject-precedes-object* constraint; 3) sequential fusion; and 4) parallel fusions (see Fig. 11). Except for the 1) among the four cases, the ordering of the subject and the object is fixed and therefore meets the directionality condition. Additionally, 3) and 4) consider the sum of all possible permutations. The difference between 3) and 4) is the order of fusion. We conduct experiments in two settings for performance comparison on fusion methods. The results are summarized in Table V. Here, the performance difference between the four fusion methods in the SGC setting [see Table V(a)] is not prominent, while the significance of combining three features is particularly evident in the BRC setting [see Table V(b)], suggesting that union feature alone cannot give relational direction. In both settings, the use of permutations at the fusion phase showed better results than otherwise, and especially when fused in parallel, it showed the best results.

4) *Effectiveness of GIH*: We examine the effectiveness of GIH by comparing GIH with two representative message

5

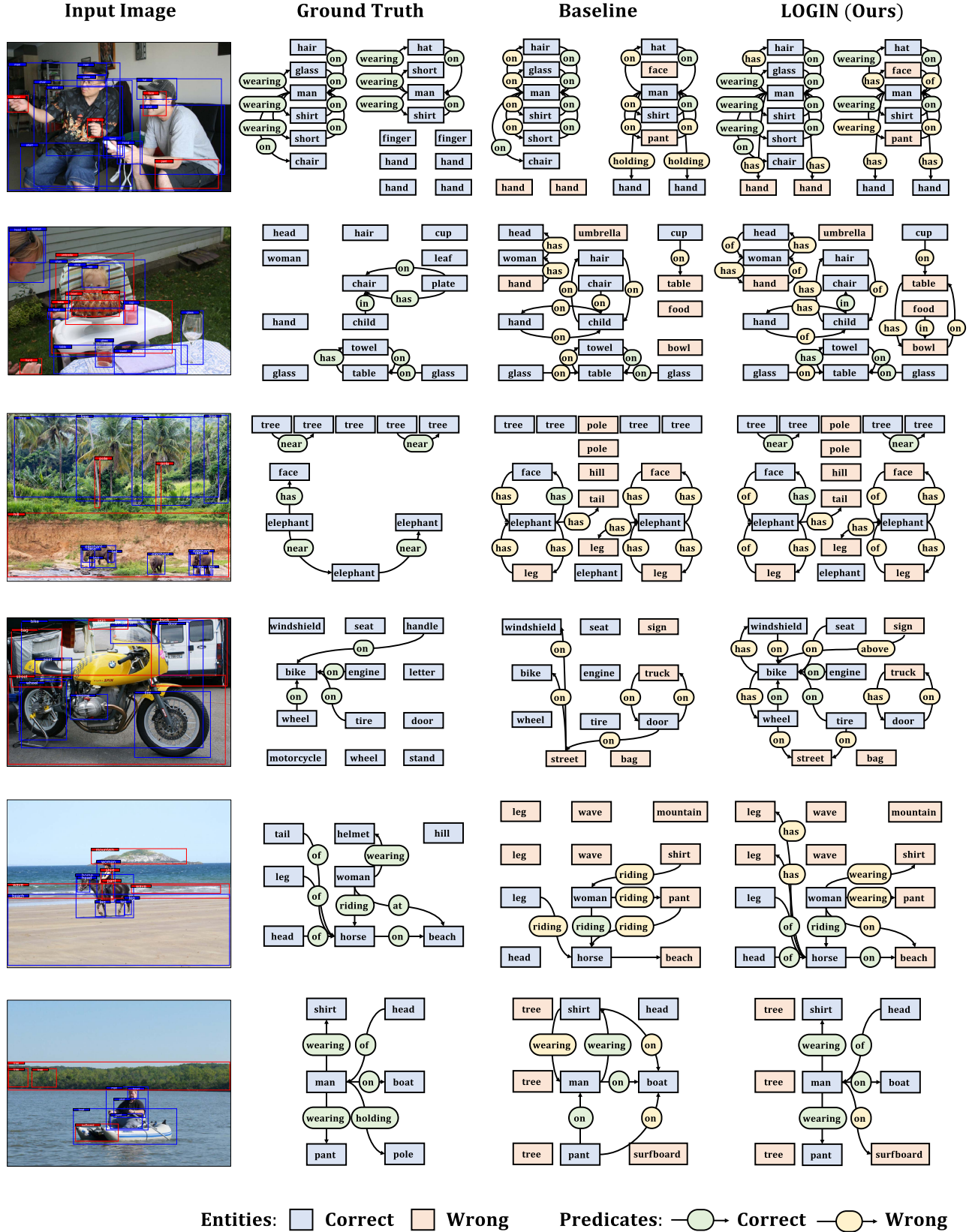


Fig. 12. Qualitative examples. The first column shows input images with entity proposals. From the second to fourth columns, we show the scene graphs of ground-truth, BASELINE, and LOGIN, respectively. The bounding boxes or nodes are colored in either blue (correct) or red (wrong). The predicates are colored in either green (correct) or yellow (wrong). Examples of the first two rows contain BRs, but not the rest. We see that LOGIN produces more diverse predicates and can successfully distinguish asymmetric relationships while BASELINE model fails.



TABLE VI  
EFFECTIVENESS OF GRAPH INTERACTION HEAD (GIH) COMPARED TO OTHER GRAPH NEURAL NETWORKS (E.G., GCN [42] AND GAT [56]). (a) SGCLS. (b) BRC

(a)				
Methods	SGCLS			
	R@20	R@50	R@100	
LOGIN /w GCN [42]	33.8	37.5	39.7	
LOGIN /w GAT [56]	33.2	37.1	39.5	
<b>LOGIN /w GIH (Ours)</b>	<b>34.5</b>	<b>38.8</b>	<b>40.5</b>	

(b)				
Fusion	PredCls			
	pR@2	pR@4	pR@8	pR@16
LOGIN /w GCN [42]	7.8	12.4	16.5	20.5
LOGIN /w GAT [56]	7.9	12.3	16.3	20.3
<b>LOGIN /w GIH (Ours)</b>	<b>8.6</b>	<b>13.1</b>	<b>17.6</b>	<b>21.1</b>

TABLE VII  
PER-TYPE PREDICATE CLASSIFICATION RESULTS. ONLY TOP-20 FREQUENT PREDICATES ARE SHOWN. THE EVALUATION METRIC IS R@50

predicate	Baseline	LOGIN	predicate	Baseline	LOGIN
on	66.3	88.1	sitting on	32.2	61.0
has	47.7	87.5	under	35.9	52.5
wearing	68.9	93.7	riding	26.3	83.0
of	42.8	82.4	in front of	8.9	29.4
in	47.7	64.1	standing on	16.7	37.7
near	19.4	52.3	at	39.5	57.8
with	18.1	45.9	attached to	12.1	21.0
behind	20.7	57.0	carrying	23.9	62.1
holding	31.4	78.7	walking on	10.5	59.3
above	18.2	51.2	over	9.5	28.4

passing graph neural networks in Table VI: Graph Convolutional Network (GCN) [42] and Graph Attention Network (GAT) [56]. GCN aggregates feature information via a non-euclidean convolution operation from a node's neighborhood. As opposed to GCNs, GAT allows for implicitly assigning different importances to nodes of the same neighborhood, enabling a leap in model capacity. Unlike them, layer-wise propagation rule of GIH considers not only nodes but also edges as a neighborhood, allowing the model to leverage higher-order contexts for node update. From the results, we see that GAT does not improve the performance upon the GCN. The results demonstrate the effectiveness of GIH in predicting both object and relationships categories (SGCLS). Since LOGIN equipped with GIH exploits richer information (e.g., edge), it is also strong in understanding relational direction (BRC).

5) *Per-Type Predicate Recall*: We expect the model to better understand each predicate by allowing attention mechanism of LIH to capture the predicate label-specific representation well and Attract and Repel loss to help separate interclass and aggregate intraclass predicates in the embedding space. In order to ensure that the proposed model solves the ambiguity issue well, we compare our LOGIN with Baseline under the Recall@50 metric for the top-20 frequent predicates in Table VII. Compared to Baseline, we observe a significant

performance improvement in all predicate classes. Specifically, our system better understands the geometric predicate (e.g., on, in front of, behind, above, under), possessive predicates (e.g., has, of, wearing), and semantic predicates (e.g., holding, walking). This suggests that explicit separation on predicate embedding space properly solves the ambiguity problem.

#### D. Qualitative Analysis

To better see how LOGIN understands the relational direction, we provide qualitative examples in Fig. 12. Here, we compare the result of BASELINE model and LOGIN with the corresponding ground-truth scene graph. As we can see in the results of first two rows, BASELINE model produces the same result for a pair of entities regardless of direction. What is worse is that the whole scene graphs use almost the same predicates for defining relationships. In other words, the BASELINE model neither considers relational-direction nor lexical diversity. On the other hand, LOGIN can successfully identify relational direction, thanks to the embedded direction-awareness, and it is also more diverse in terms of vocabulary. More interestingly, even though predictions of LOGIN are not matched, the results are seemingly plausible. For example, in the third row, detected tail, legs, and face of an elephant are false positives in terms of ground-truth, but they seem to be correct in reality. Also, relationships associated with false positives are somewhat reasonable (e.g., elephant  $\xrightarrow{\text{has}}$  leg, leg  $\xrightarrow{\text{of}}$  elephant).

## VI. CONCLUSION

This article discusses three fundamental challenges in SGG task: 1) ambiguity; 2) asymmetry; and 3) higher-order contexts. Motivated by the analysis and to tackle the issues effectively, we present a new unified framework, LOGIN. Our framework enables predicting the scene graph in a local-to-global and bottom-to-up manner, leveraging the possible complementarity effectively. We achieved state-of-the-art on VG benchmark. Last but not least, we present a new diagnostic task called BRC and observe that our method outperforms competing methods significantly.

## REFERENCES

- [1] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.
- [2] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. Int. Conf. Comput. Vis. (CVPR)*, 2017, pp. 2961–2969.
- [3] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.
- [4] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Dec. 2016, pp. 779–788.
- [5] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
- [6] J. Johnson *et al.*, "Image retrieval using scene graphs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3668–3678.
- [7] D. Teney, L. Liu, and A. Van Den Hengel, "Graph-structured representations for visual question answering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1–9.
- [8] T. Yao, Y. Pan, Y. Li, and T. Mei, "Exploring visual relationship for image captioning," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 684–699.

- [9] J. Johnson, A. Gupta, and L. Fei-Fei, "Image generation from scene graphs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 1219–1228.
- [10] C.-Y. Ma, A. Kadav, I. Melvin, Z. Kira, G. AlRegib, and H. P. Graf, "Attend and interact: higher-order object interactions for video understanding," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6790–6800.
- [11] X. Yang, K. Tang, H. Zhang, and J. Cai, "Auto-encoding scene graphs for image captioning," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 10685–10694.
- [12] O. Ashual and L. Wolf, "Specifying object attributes and relations in interactive scene generation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4561–4569.
- [13] Z. Li, Q. Tran, L. Mai, Z. Lin, and A. Yuille, "Context-aware group captioning via self-attention and contrastive features," 2020, *arXiv:2004.03708*.
- [14] R. Krishna *et al.*, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *Int. J. Comput. Vis.*, vol. 123, no. 1, pp. 32–73, 2017.
- [15] R. Hu, M. Rohrbach, J. Andreas, T. Darrell, and K. Saenko, "Modeling relationships in referential expressions with compositional modular networks," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 1115–1124.
- [16] G. Gkioxari, R. Girshick, P. Dollar, and K. He, "Detecting and recognizing human-object interactions," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8359–8367.
- [17] Y.-W. Chao, Y. Liu, X. Liu, H. Zeng, and J. Deng, "Learning to detect human-object interactions," in *Proc. Winter Conf. Appl. Comput. Vis. (WACV)*, 2018, pp. 381–389.
- [18] Y.-L. Li *et al.*, "Transferable interactiveness knowledge for human-object interaction detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3585–3594.
- [19] C. Lu, R. Krishna, M. Bernstein, and L. Fei-Fei, "Visual relationship detection with language priors," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Amsterdam, The Netherlands: Springer, 2016, pp. 852–869.
- [20] H. Zhang, Z. Kyaw, S.-F. Chang, and T.-S. Chua, "Visual translation embedding network for visual relation detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5532–5540.
- [21] Y. Li, W. Ouyang, X. Wang, and X. Tang, "ViP-CNN: Visual phrase guided convolutional neural network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1347–1356.
- [22] B. A. Plummer, A. Mallya, C. M. Cervantes, J. Hockenmaier, and S. Lazebnik, "Phrase localization and visual relationship detection with comprehensive image-language cues," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Jun. 2017, pp. 1928–1937.
- [23] H. Zhang, Z. Kyaw, J. Yu, and S.-F. Chang, "PPR-FCN: Weakly supervised visual relation detection via parallel pairwise R-FCN," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, 2017, pp. 4233–4241.
- [24] B. Dai, Y. Zhang, and D. Lin, "Detecting visual relationships with deep relational networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3076–3086.
- [25] X. Liang, L. Lee, and E. P. Xing, "Deep variation-structured reinforcement learning for visual relationship and attribute detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 848–857.
- [26] L. Mi and Z. Chen, "Hierarchical graph attention network for visual relationship detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 13886–13895.
- [27] D. Xu, Y. Zhu, C. B. Choy, and L. Fei-Fei, "Scene graph generation by iterative message passing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5410–5419.
- [28] R. Zellers, M. Yatskar, S. Thomson, and Y. Choi, "Neural motifs: Scene graph parsing with global context," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5831–5840.
- [29] S. Woo, D. Kim, D. Cho, and I. S. Kweon, "LinkNet: Relational embedding for scene graph," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2018, pp. 560–570.
- [30] J. Yang, J. Lu, S. Lee, D. Batra, and D. Parikh, "Graph R-CNN for scene graph generation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 670–685.
- [31] Y. Li, W. Ouyang, B. Zhou, J. Shi, C. Zhang, and X. Wang, "Factorizable net: An efficient subgraph-based framework for scene graph generation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 335–351.
- [32] M. Qi, W. Li, Z. Yang, Y. Wang, and J. Luo, "Attentive relational networks for mapping images to scene graphs," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 3957–3966.
- [33] T. Chen, W. Yu, R. Chen, and L. Lin, "Knowledge-embedded routing network for scene graph generation," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6163–6171.
- [34] K. Tang, H. Zhang, B. Wu, W. Luo, and W. Liu, "Learning to compose dynamic tree structures for visual contexts," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6619–6628.
- [35] L. Chen, H. Zhang, J. Xiao, X. He, S. Pu, and S.-F. Chang, "Counterfactual critic multi-agent training for scene graph generation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Mar. 2019, pp. 4613–4623.
- [36] J. Zhang, K. J. Shih, A. Elgammal, A. Tao, and B. Catanzaro, "Graphical contrastive losses for scene graph parsing," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 11535–11543.
- [37] A. Zareian, S. Karaman, and S.-F. Chang, "Weakly supervised visual semantic parsing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 3736–3745.
- [38] K. Tang, Y. Niu, J. Huang, J. Shi, and H. Zhang, "Unbiased scene graph generation from biased training," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2020, pp. 3716–3725.
- [39] A. Zareian, S. Karaman, and S.-F. Chang, "Bridging knowledge graphs to generate scene graphs," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 606–623.
- [40] W. Wang, R. Wang, S. Shan, and X. Chen, "Sketching image gist: Human-mimetic hierarchical scene graph generation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 222–239.
- [41] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2015, pp. 91–99.
- [42] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2017, pp. 1–14.
- [43] X. Yang, H. Zhang, and J. Cai, "Shuffle-then-assemble: Learning object-agnostic visual relationship features," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 36–52.
- [44] Y. Li, W. Ouyang, B. Zhou, K. Wang, and X. Wang, "Scene graph generation from objects, phrases and region captions," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, 2017, pp. 1261–1270.
- [45] J. Fu, H. Zheng, and T. Mei, "Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 4438–4446.
- [46] H. Zheng, J. Fu, T. Mei, and J. Luo, "Learning multi-attention convolutional neural network for fine-grained image recognition," in *Proc. Int. Conf. Comput. Vis. (ICCV)*, 2017, pp. 5209–5217.
- [47] X.-S. Wei, C.-W. Xie, J. Wu, and C. Shen, "Mask-CNN: Localizing parts and selecting descriptors for fine-grained bird species categorization," *Pattern Recognit.*, vol. 76, pp. 704–714, Apr. 2018.
- [48] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7794–7803.
- [49] S. Woo, J. Park, J.-Y. Lee, and I. So Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19.
- [50] H. Zheng, J. Fu, Z.-J. Zha, and J. Luo, "Looking for the devil in the details: Learning trilinear attention sampling network for fine-grained image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 5012–5021.
- [51] H. Wang, Y. Zhu, B. Green, H. Adam, A. Yuille, and L.-C. Chen, "Axial-deeplab: Stand-alone axial-attention for panoptic segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 108–126.
- [52] A. Santoro *et al.*, "A simple neural network module for relational reasoning," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 4967–4976.
- [53] M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. R. Salakhutdinov, and A. J. Smola, "Deep Sets," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 3391–3401.
- [54] Y. Zhang, J. Hare, and A. Prugel-Bennett, "Deep set prediction networks," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2019, pp. 3212–3222.
- [55] J. Lee, Y. Lee, J. Kim, A. Kosiorek, S. Choi, and Y. W. Teh, "Set transformer: A framework for attention-based permutation-invariant neural networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2019, pp. 3744–3753.
- [56] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 1–12.

- [57] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *Proc. 34th Int. Conf. Mach. Learn. (ICML)*, 2017, pp. 1263–1272.
- [58] G. Li, M. Müller, A. Thabet, and B. Ghanem, "DeepGCNs: Can GCNs go as deep as CNNs?" in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9267–9276.
- [59] R. Liu *et al.*, "An intriguing failing of convolutional neural networks and the coordconv solution," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2018, pp. 9605–9616.



**Sangmin Woo** (Student Member, IEEE) received the B.S. degree in electrical engineering from Kyungpook National University, Daegu, South Korea, in 2019, and the M.S. degree in electrical engineering and computer science from the Gwangju Institute of Science and Technology (GIST), Gwangju, South Korea, in 2021. He is currently pursuing the Ph.D. degree in electrical engineering with the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea.

His research interests lie in computer vision and machine learning, especially in a high-level visual understanding.



**Junhyug Noh** (Member, IEEE) received the B.S. degree in computer science and engineering, and statistics, and the M.S. and Ph.D. degrees in computer science engineering from Seoul National University, Seoul, South Korea, in 2013, 2015, and 2020, respectively.

He is currently a Post-Doctoral Researcher with the Lawrence Livermore National Laboratory (LLNL), Livermore, CA, USA. His research has focused on artificial intelligence, machine learning, and computer vision with a particular interest in object detection and its related high-level vision tasks, such as semantic/instance segmentation, scene understanding, and image captioning.



**Kangil Kim** (Member, IEEE) received the B.S. degree in computer science from the Korea Advanced Institute of Science and Technology, Daejeon, South Korea, in 2006, and the Ph.D. degree from Seoul National University, Seoul, South Korea, in 2012.

He was a Senior Researcher with the Natural Language Processing Group, Electronics and Telecommunications Research Institute, Seoul, until 2016, and an Assistant Professor with the Department of Computer Science and Engineering, Konkuk University, Seoul, until 2019. He is currently an Assistant Professor with the Electronics Engineering and Computer Science Department and the Artificial Intelligence Graduate School, Gwangju Institute of Science and Technology, Gwangju, South Korea. His research interests include artificial intelligence, evolutionary computation, machine learning, and natural language processing.