Generating Fine-Grained Scene Graphs via Relational Semantic Enhancement Networks

1st Yantao Shang
School of Computer Engineering and Science
Shanghai University
Shanghai, China
2118537431@shu.edu.cn

2nd Liyan Ma
School of Computer Engineering and Science
Shanghai University
Shanghai, China
liyanma@shu.edu.cn

3rd Xiangfeng Luo
School of Computer Engineering and Science
Shanghai University
Shanghai, China
luoxf@shu.edu.cn

4th Shaorong Xie*
School of Computer Engineering and Science
Shanghai University
Shanghai, China
srxie@shu.edu.cn

5th Wendi Rao
School of Computer Engineering and Science
Shanghai University
Shanghai, China
raowendi@shu.edu.cn

6th Chao Ma
School of Computer Engineering and Science
Shanghai University
Shanghai, China
machao23@shu.edu.cn

***Abstract*—Scene graphs, with their highly structured visual content representation, are widely applied in downstream tasks (e.g., visual question answering and human-computer interaction) and have become indispensable for scene understanding. However, existing models often struggle with long-tailed distributions and coarse-grained labeling, leading to uninformative predicate predictions such as "on" or "has". To mitigate this limitation, we introduce the Relational Semantic Enhancement Network (RSEN), which generates refined scene graphs with enhanced semantic depth. RSEN integrates three complementary modules: specifically, the Hybrid Visual Alignment module, which supplements CLIP's semantic cues with fine-grained spatial information derived from convolutional layers via cross-attention; the Hierarchical Context Interaction module, which uses a gated mechanism paired with confidence feedback to aggregate global contextual relationships, effectively suppressing relational noise; and the Semantic Prototype Refinement module, which reconstructs the feature space through the CLIP semantic prior, achieving prototype-guided metric learning and thereby capturing fine-grained semantic details. Experiments conducted on the Visual Genome and GQA datasets have demonstrated that RSEN significantly outperforms existing mainstream methods. The semantically rich scene graphs generated by RSEN can provide strong support for downstream scene understanding tasks.**

***Index Terms*—category-specific metric learning, fine-grained predicates, long-tailed distribution, scene graph generation.**

I. Introduction

Broadly speaking, scene graphs [1]–[5] represent object relationships as "entity-predicate-entity" triplets, providing structured semantic interfaces for downstream tasks such as visual question answering [6] and image retrieval [7]. Despite its great potential, extracting unbiased and semantically rich scene graphs remains a significant challenge due to the complexity of relational reasoning.

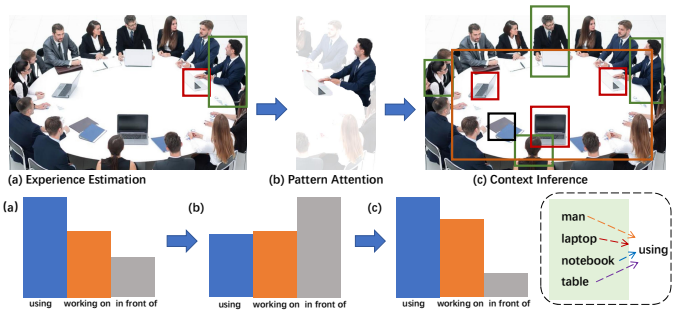

Fig. 1. Global context aids in scene disambiguation but lacks fine-grained discriminative power. While surrounding objects help identify the general activity, existing models still struggle to differentiate subtle semantic variations, often resulting in uninformative predictions like "using".

In fact, obtaining unbiased and semantically rich scene graphs requires a sophisticated integration of multi-level information. As illustrated by the meeting scenario in Fig. 1. Human reasoning for relationship prediction typically follows a tripartite logic: first, statistical priors that involve recognizing that "people" and "laptops" often imply a "using" relationship; second, local associations that focus on specific patterns like "hand near keyboard", which might lead to a prediction of "in front of"; finally, global context that utilizes surrounding objects (e.g., notebooks, round tables) to disambiguate the scene as a "conference", thereby suppressing inappropriate predictions like "in front of". However, even with such context, models often still fail to achieve semantic depth, defaulting to the high-frequency but uninformative predicate "using" instead of the more precise "working on".

Motivated by these observations, we argue that current limitations stem from two main challenges: On one hand, tradi-

tional message-passing mechanisms often treat all associations equally, and struggle to filter out irrelevant scene noise or prioritize critical global context. On the other hand, even with contextual cues, existing models lack sufficient semantic depth, unable to distinguish fine-grained variations within similar predicate categories.

Based on this background, we propose the Relational Semantic Enhancement Network (RSEN), which refines scene graph generation through three synergistic stages. First, the Hybrid Visual Alignment (HVA) module leverages CLIP [8] priors to establish a deep semantic foundation. This is followed by the Hierarchical Context Interaction (HCI) module, which aggregates contextual information through a gated bipartite graph and filters out noise. Finally, the Semantic Prototype Refinement (SPR) module projects these features into a prototype-based metric space, adaptively resolving ambiguities between visually similar predicates.

The main contributions of our works are three-folds:

- **Hybrid Visual Alignment**: We design the HVA module, aiming to combine the visual embeddings of CLIP with convolutional features, thereby effectively transferring the general visual language prior knowledge to the scene graph generation task.
- **Hierarchical Context Interaction**: We design a bidirectional closed-loop mechanism to regulate information flow between entities and predicates, and leverage confidence feedback mechanisms to mitigate information overload in contextual propagation.
- **Semantic Prototype Refinement**: We design a category-specific metric learning framework with dimensional attention, which extracts a minimal discriminative feature set for each predicate and notably boosts the fine-grained separability of the semantic feature space.

II. Related Work

Scene Graph Generation (SGG): SGG aims to bridge the gap between visual perception and semantic understanding by parsing images into structured graphs. Early methods such as Motifs [2] relied on statistical priors and global context pooling. Xu *et al.* [4] proposed iterative message passing between nodes and edges; the cyclic structure jointly optimizes object and predicate features. In recent years, the bipartite graph architecture (such as BGNN [9]) has gained significant attention due to its advantages in entity-predicate modeling. Its bidirectional interaction and adaptive mechanisms provide a new approach for handling contextual noise.

Unbiased Scene Graph Generation: To address the biased prediction problem, researchers have adopted various methods. Among rebalancing-based approaches, HTCL [10] resamples the training set to rebalance class frequencies. For loss function optimization, Class-Balanced Loss [7] re-weights each category so that the model pays more attention to tail classes. Recent research has shifted its focus to cross-category knowledge transfer techniques: PENet [11] and DPL [12] link head and tail via prototype learning; TDE [13] removes co-occurrence bias with causal intervention, thus purifies the model's reasoning logic.

III. Method

Our scene graph generation model is a modular deep network consisting of three cascaded submodules. As illustrated in Fig. 2, RSEN consists of three stages: Hybrid Visual Alignment (HVA), Hierarchical Context Interaction (HCI), and Semantic Prototype Refinement (SPR). The details of these stages are provided in the following subsections.

A. Problem Setting

Given an image $\mathcal{I}$, SGG aims to generate a structured graph $\mathcal{G} = \langle \mathcal{O}, \mathcal{E} \rangle$. Here, $\mathcal{O} = \{o_i\}_{i=1}^{N}$ denotes the set of detected objects with categories C and bounding boxes $\mathcal{B}$. $\mathcal{E} = \{e_k\}_{k=1}^{M}$ represents the set of edges, where each $e_k = (o_i, r_k, o_j)$ defines a triplet with a predicate r_k between subject o_i and object o_j.

B. Hybrid Visual Alignment Module

Following the standard pipeline [13], we extract object candidates $\mathcal{O} = \{o_i\}$ and their union-region boxes. For each region, we extract a base semantic embedding $\mathbf{z} \in \mathbb{R}^d$ from a pre-trained CLIP visual encoder. To compensate for CLIP's lack of localized spatial precision, we derive a complementary convolutional feature $\mathbf{o}_i$ for each entity by integrating its visual, spatial, and category cues via a projection f_o:

$$\mathbf{o}_i = f_o(\mathbf{v}_i \oplus \mathbf{b}_i \oplus \mathbf{l}_i), \quad \mathbf{u}_{i\to j} = f_u(\mathbf{o}_i \oplus \mathbf{o}_j), \tag{1}$$

where $\mathbf{o}_i, \mathbf{u}_{i\to j} \in \mathbb{R}^d$ represent the localized entity and union-region cues, respectively.

To integrate local spatial information into the CLIP embedding, we employ a cross-attention mechanism to fuse the CLIP embedding and the convolutional features:

$$\mathrm{Attn}(\mathbf{Q}, \mathbf{K}) = \mathrm{softmax}\left(\frac{(\mathbf{Q}\mathbf{W}^{Q})(\mathbf{K}\mathbf{W}^{K})^{T}}{\sqrt{d}}\right)(\mathbf{K}\mathbf{W}^{V}), \tag{2}$$

where $\mathbf{W}^{Q}, \mathbf{W}^{K}, \mathbf{W}^{V} \in \mathbb{R}^{d\times d}$ are learnable projection matrices. By treating the CLIP embeddings as Queries ($\mathbf{Q}$) and the convolutional features as Keys ($\mathbf{K}$) and Values ($\mathbf{V}$), this module adaptively extracts spatial details. The initial representations of entities and predicates are then calculated as:

$$\mathbf{x}_i = \mathrm{LayerNorm}(\mathbf{z}_i + \mathrm{Attn}(\mathbf{z}_i, \mathbf{o}_i)), \tag{3}$$

$$\mathbf{e}_{i\to j} = \mathrm{LayerNorm}(\mathbf{z}_{i\to j} + \mathrm{Attn}(\mathbf{z}_{i\to j}, \mathbf{u}_{i\to j})), \tag{4}$$

where $\mathbf{x}_i$ and $\mathbf{e}_{i\to j}$ represent the final features that possess both semantic depth and spatial foundation. To preserve the original CLIP features during the fusion process, we employ residual connections and perform normalization.

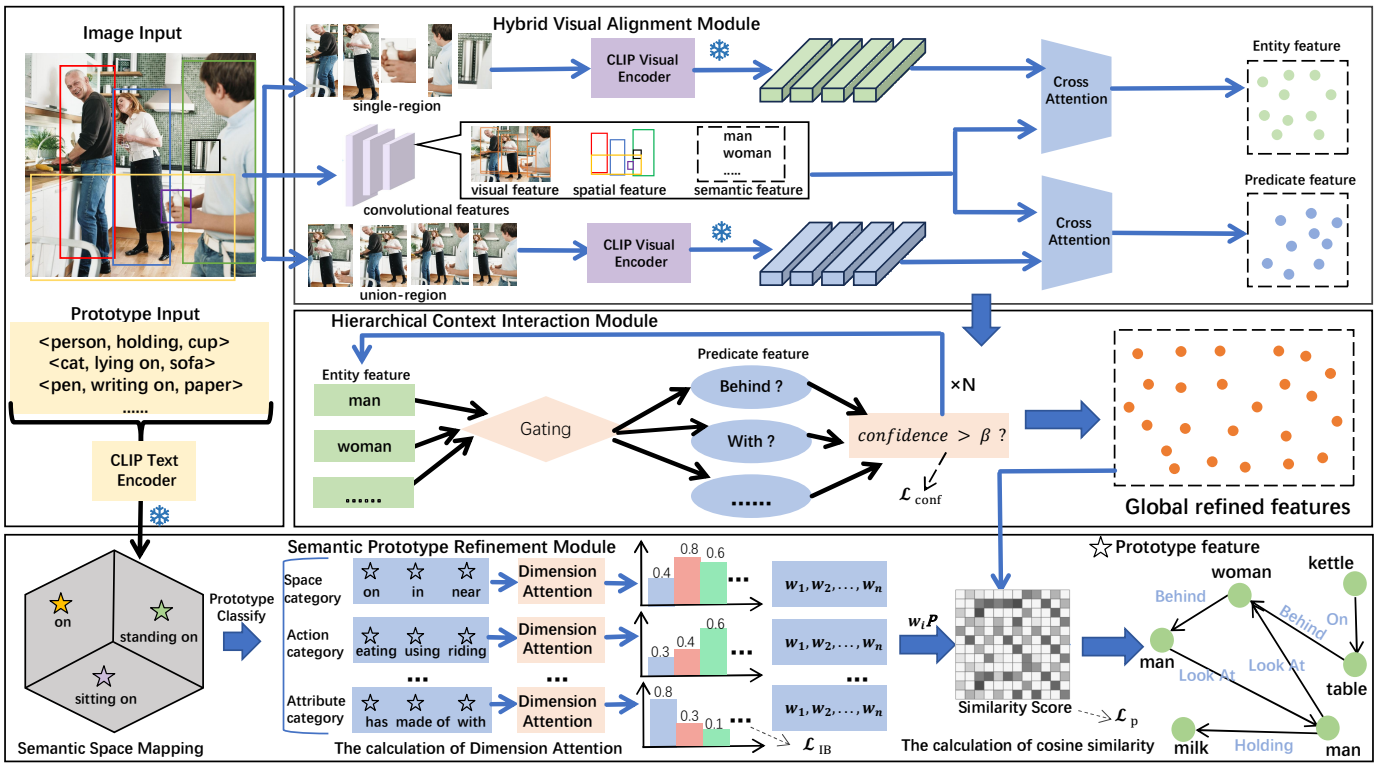

Fig. 2. The workflow of RSEN. The framework comprises: (1) HVA for fusing CLIP semantic priors with convolutional features via cross-attention; (2) HCI for filtering relational noise through a bidirectional gated mechanism and confidence feedback; and (3) SPR for resolving predicate ambiguities using dimension-wise attention over semantic prototypes. The final predicatea is generated via prototype matching in a re-weighted feature space.

C. Hierarchical Context Interaction Module

To aggregate context while filtering noise, we perform bidirectional message passing on a bipartite graph. We take the aligned features $\mathbf{x}_i$ and $\mathbf{e}_{i\to j}$ from the HVA module as the initial states $\mathbf{x}_i^0$ and $\mathbf{e}_{i\to j}^0$.

Entity → Predicate Gated Transmission: Unlike traditional methods that treat all entities equally, we introduce a gating coefficient g_k to regulate the influence of global entities based on their semantic relevance to a specific predicate:

$$g_k = \sigma\left(\mathbf{W}_g^\top [\mathbf{e}_{i\to j}^l \oplus \mathbf{x}_k^l]\right), \tag{5}$$

$$\mathbf{e}_{i\to j}^{l+1} = \mathbf{e}_{i\to j}^l + \mathrm{ReLU}\left(\sum_{k=1}^{N} g_k \cdot \mathbf{W}_e^\top \mathbf{x}_k^l\right), \tag{6}$$

where $\mathbf{W}_g$ and $\mathbf{W}_e$ are learnable weights, and N is the total number of entity nodes. This ensures that only relevant entities (e.g., "tire" for "attached to") contribute to the refinement of the predicate.

Predicate → Entity Confidence Feedback: In order to improve the representation of entities with reliable relational contexts, we first estimate the predicate confidence score $s_{i\to j}^l$:

$$s_{i\to j}^l = \sigma(\mathbf{W}_s^\top [\mathbf{e}_{i\to j}^l \oplus \mathbf{x}_i \oplus \mathbf{x}_j]), \tag{7}$$

Then, a gating function is used to map the confidence score and suppress the noise:

$$G(s) = \begin{cases} 0, & s \le \beta, \\ \alpha(s-\beta), & \beta < s < \gamma, \\ 1, & s \ge \gamma, \end{cases} \tag{8}$$

where learnable thresholds α, β, γ partition predicates into noisy, ambiguous, and reliable samples. Predicates with too low confidence are eliminated, and then the entity features are updated by aggregating the feedback of predicates associated with each entity:

$$\mathbf{x}_i^{l+1} = \mathbf{x}_i^l + \varphi\left(\frac{1}{|\Phi(o_i)|}\sum_{p\in\Phi(o_i)} G(s_p^l)\mathbf{W}_p^\top \mathbf{e}_p^l\right), \tag{9}$$

where $\Phi(o_i)$ is the set of all predicates involving entity node i, $\mathbf{W}_p \in \mathbb{R}^{d\times d}$ is the linear transformation matrix from predicate features to entity features, and $G(\cdot)$ is the gating coefficient.

To ensure the reliability of the feedback gating, the predicted confidence score s is constrained by a binary cross-entropy loss $\mathcal{L}_{\text{conf}}$ against the ground-truth $\hat{s} \in \{0, 1\}$(where 1 denotes a valid relationship and 0 otherwise).

D. Semantic Prototype Refinement Module

Although the HCI module yields context-rich features, simple label mapping often fails to address the polysemy of predicates and the severe class imbalance. We thus propose the Semantic

Prototype Refinement (SPR) module to achieve the prediction of fine-grained predicates.

Semantic Space Mapping: To capture contextual nuances, we encode triplet-based prompts (e.g., "[Sub] is [Pred] [Obj]") sampled from the datasets. These prompts are then fed into a pre-trained CLIP text encoder to extract context-aware embeddings. Finally, the prototype $\mathcal{P} = \{\mathbf{p}_k\}_{k=1}^{n}$ is the average embedding of all corresponding prompts for category k.

The calculation of Dimension Attention: While CLIP prototypes provide semantic foundations, standard cosine similarity treats dimensions uniformly, and overlooks category-specific feature patterns. To address this, we introduce a Dimensional Attention mechanism to learn prototype-specific weights $\boldsymbol{\omega}_i$. We first partition all predicate prototypes into several semantic groups G based on attributes (e.g., spatial or action). For any group G containing n prototypes, we construct a prototype matrix $\mathbf{P}_G = [\mathbf{p}_1, \ldots, \mathbf{p}_n]^\top \in \mathbb{R}^{n\times d}$. $\mathbf{P}_G$ is projected into query ($\mathbf{Q}$), key ($\mathbf{K}$), and value ($\mathbf{V}$) spaces:

$$\mathbf{Q} = \mathbf{P}_G\mathbf{W}_q, \quad \mathbf{K} = \mathbf{P}_G\mathbf{W}_k, \quad \mathbf{V} = \mathbf{P}_G\mathbf{W}_v \tag{10}$$

where $\mathbf{W}_q, \mathbf{W}_k, \mathbf{W}_v \in \mathbb{R}^{d\times d}$ are learnable weights. We then compute a similarity weight matrix $A \in \mathbb{R}^{n\times n}$ via scaled dot-product attention:

$$\mathbf{A} = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d}}\right), \tag{11}$$

where A_{ij} represents the semantic reliance of prototype i on prototype j. By leveraging $\mathbf{A}$ to aggregate $\mathbf{V}$, each prototype perceives the feature distribution of its peers. The aggregated features are then processed through a MLP and a Sigmoid function σ to generate the final dimensional attention vector $\mathbf{w}_i$:

$$\mathbf{w}_i = \sigma\left(\text{MLP}\left(\sum_{j=1}^{n} A_{ij}\cdot\mathbf{V}_j\right)\right) \in \mathbb{R}^d, \tag{12}$$

where $\mathbf{V}_j$ denotes the j-th row vector of matrix $\mathbf{V}$. Each dimension of $\mathbf{w}_i \in (0, 1)$ signifies its discriminative importance. Values approaching 1 identify core dimensions essential for distinguishing similar predicates within the group, while values near 0 suppress redundant or noisy dimensions.

To ensure the effectiveness of this Dimensional Attention, we design a dual-constraint regularization loss $\mathcal{L}_{reg}$:

$$\mathcal{L}_{\text{reg}} = \lambda_s \sum_{d=1}^{D} |\omega_{i,d}| - \sum_{d=1}^{D} \bar{\omega}_{i,d} \log \bar{\omega}_{i,d}, \tag{13}$$

where $d \in \{1, \ldots, D\}$ denotes the feature dimension index, and $\bar{\omega}_{i,d} = \text{Softmax}(\omega_{i,d})$ represents the normalized attention weight. Specifically, $\mathcal{L}_{\text{reg}}$ refines the relational metric space through two synergistic constraints: The L_1-norm term enforces structural sparsity, effectively pruning redundant or noisy dimensions that do not contribute to predicate discrimination; The second term performs entropy minimization (note the negative sign inherent in entropy definition, here formulated to be minimized), which encourages the attention distribution to be peaked rather than uniform.

Without these constraints, semantically similar predicates (e.g., "sitting on" vs."standing on") might yield overlapping attention patterns, rendering the model unable to distinguish their subtle nuances.

The final classification probability is calculated based on the modulation cosine similarity of the specific subspace of the features:

$$p(i \mid \mathbf{x}_i) = \text{Softmax}\left(\alpha \cdot \frac{(\mathbf{x}_i \odot \mathbf{w}_i)\cdot(\mathbf{p}_i \odot \mathbf{w}_i)^\top}{|\mathbf{x}_i \odot \mathbf{w}_i|_2 |\mathbf{p}_i \odot \mathbf{w}_i|_2}\right) \tag{14}$$

where $\mathbf{x}_i \in \mathbb{R}^d$ denotes the predicate features obtained from the previous module, and α is a learnable scaling factor.

E. Optimization

We train the model in an end-to-end framework by optimizing the sum of all loss functions with different weights in each module. The loss function of the entire model can be refined into:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{p}} + \lambda_{\text{conf}}\mathcal{L}_{\text{conf}} + \lambda_{\text{reg}}\mathcal{L}_{\text{reg}}, \tag{15}$$

where $\mathcal{L}_{\text{p}}$ denote the cross-entropy loss for predicate classification. The coefficient λ_{conf} and λ_{reg} balance the predicate confidence score and the dimensional attention regularization, respectively.

IV. Experiments

A. Datasets

We primarily evaluate RSEN on the Visual Genome (VG) dataset [22]. Following the widely adopted split [2], we consider the most frequent 150 object classes and 50 predicate classes. This subset contains approximately 108k images, split into a training set (70%, ~75k) and a test set (30%, ~32k). We further conduct additional experiments on the GQA dataset [23] to verify the model robustness.

B. Implementation Details

Following [13], we adopt ResNeXt-101-FPN [24] as the backbone and Faster R-CNN [25] as the detector. For VG, the pre-trained detector from [3] is employed for a fair comparison; for GQA, we train a customized detector with the same architecture as no standard version is publicly available. The CLIP model and the detector are frozen during training. RSEN is optimized via the Adam with a learning rate of 10^{-4} and a batch size 12. In the HCI module, thresholds (α, β, γ) are initialized at $(0.50, 0.25, 0.75)$ and adaptively optimized. Due to space constraints, we report the optimal loss weights determined via empirical validation: $\lambda_{\text{conf}} = 0.5$ and $\lambda_{\text{reg}} = 0.3$. Within the dual-constraint loss $\mathcal{L}_{\text{reg}}$, the coefficient for the sparsity term is fixed at $\lambda_s = 0.1$.

C. Tasks and Metrics

We evaluate RSEN on three subtasks: (1) *PredCls*: predicting predicates given ground-truth boxes and object classes; (2) *SGCls*: predicting object classes and predicates given ground-truth boxes; (3) *SGGen*: detecting objects and predicting predicates from raw images. Recall@K (R@K) and mean Recall@K (mR@K) for $K \in \{50, 100\}$ are used as evaluation metrics.

TABLE I
PERFORMANCE COMPARISON ON VG AND GQA DATASETS (%).

Dataset	Model	PredCls		SGCls		SGGen	
		mR@50/100	R@50/100	mR@50/100	R@50/100	mR@50/100	R@50/100
VG	Motifs [2]	14.8 / 16.1	65.2 / 67.0	8.3 / 8.8	38.9 / 39.8	6.8 / 7.9	32.9 / 37.2
	+TDE [13]	25.5 / 29.1	46.2 / 51.4	13.1 / 14.9	27.7 / 29.9	8.2 / 9.8	16.9 / 20.3
	+PCPL [14]	24.3 / 26.1	54.7 / 56.5	12.0 / 12.7	35.3 / 36.1	10.7 / 12.6	27.8 / 31.7
	+CogTree [15]	26.4 / 29.0	35.6 / 36.8	14.9 / 16.1	21.6 / 22.2	10.4 / 11.8	20.0 / 22.1
	+CAModule [16]	36.7 / 39.3	59.8 / 63.4	21.1 / 24.7	36.8 / 38.2	16.3 / 18.2	29.1 / 32.7
	VTransE [17]	14.7 / 15.8	65.7 / 67.6	8.2 / 8.7	38.6 / 39.4	5.0 / 6.0	29.7 / 34.3
	+TDE [13]	24.6 / 28.0	43.1 / 48.5	12.9 / 14.8	25.7 / 28.5	8.6 / 10.5	18.7 / 22.6
	+DPL [12]	33.3 / 36.3	56.2 / 58.0	16.3 / 18.2	30.4 / 32.9	11.2 / 13.7	19.4 / 24.2
	VCTree [3]	16.7 / 18.2	65.4 / 67.2	11.8 / 12.5	46.7 / 47.6	7.4 / 8.7	31.9 / 36.2
	+TDE [13]	25.4 / 28.7	47.2 / 51.6	12.2 / 14.0	25.4 / 27.9	9.3 / 11.1	19.4 / 23.2
	+PCPL [14]	22.8 / 24.5	56.9 / 58.7	15.2 / 16.1	40.6 / 41.7	10.8 / 12.6	26.6 / 30.3
	+CogTree [15]	26.4 / 29.0	35.6 / 36.8	14.9 / 16.1	21.6 / 22.2	10.4 / 11.8	20.0 / 22.1
	+CAModule [16]	38.4 / 40.5	60.6 / 63.8	23.2 / 25.9	38.6 / 40.2	16.4 / 19.6	29.8 / 33.4
	BGNN [9]	30.4 / 32.9	59.2 / 61.3	14.3 / 16.5	37.4 / 38.5	10.7 / 12.6	31.0 / 35.8
	GPS-Net [18]	15.2 / 16.6	65.2 / 67.1	8.5 / 9.1	37.8 / 39.2	6.7 / 8.6	31.1 / 35.9
	KERN [19]	17.7 / 19.2	65.8 / 67.6	9.4 / 10.0	36.7 / 37.4	6.4 / 7.3	27.1 / 29.8
	RSFA [20]	38.0 / 40.4	48.1 / 50.3	21.2 / 22.2	28.5 / 29.4	14.8 / 17.4	23.4 / 27.9
	EdgeSGG [21]	34.7 / 36.9	60.1 / 61.8	17.8 / 18.8	39.1 / 40.1	14.6 / 16.9	29.7 / 34.0
	RSEN(Ours)	**39.1 / 40.5**	57.1 / 60.1	**23.3 / 26.8**	39.5 / 40.3	**19.1 / 20.0**	25.5 / 28.1
GQA	Motifs [2]	14.9 / 15.6	64.9 / 66.4	6.2 / 6.5	36.1 / 37.9	4.2 / 5.2	27.7 / 32.3
	VTransE [17]	14.9 / 15.8	65.0 / 66.5	6.9 / 7.3	35.2 / 36.1	5.3 / 6.2	31.7 / 33.6
	VCTree [3]	15.7 / 16.6	65.0 / 66.6	6.1 / 6.5	35.8 / 42.9	4.1 / 5.1	28.6 / 34.5
	BGNN [9]	27.4 / 30.3	59.1 / 60.9	13.6 / 15.1	36.5 / 37.8	9.7 / 12.9	30.2 / 31.8
	GPS-Net [18]	15.3 / 17.0	64.2 / 66.0	8.7 / 9.4	36.4 / 37.5	7.7 / 9.1	30.1 / 33.0
	KERN [19]	16.8 / 18.8	63.8 / 64.9	9.8 / 10.7	35.4 / 36.8	6.9 / 7.7	25.7 / 27.8
	RSFA [20]	36.2 / 37.4	46.1 / 50.3	20.3 / 22.1	27.2 / 29.6	14.3 / 17.1	22.4 / 25.9
	EdgeSGG [21]	33.7 / 36.9	57.6 / 60.3	17.3 / 17.6	38.5 / 39.4	13.6 / 15.2	28.6 / 33.5
	RSEN(Ours)	**34.1 / 37.5**	56.1 / 57.7	**22.3 / 23.7**	37.5 / 38.1	**17.9 / 19.1**	24.6 / 27.2

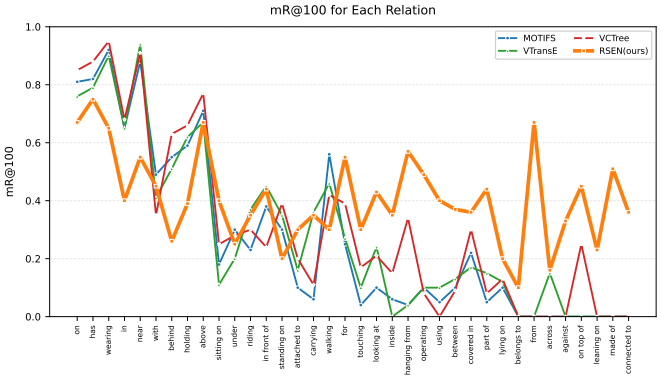

Fig. 3. mR@100 performance comparison between other methods and RSEN (Ours) for each relation type on the VG dataset.

TABLE II
ABLATION STUDY OF RSEN MODEL COMPONENTS (MR@100 METRIC) ON VG.

Module			mR@100		
HVA module	HCI module	SPR module	PredCls	SGCls	SGGen
×	×	×	11.8	6.7	4.1
✓	×	×	25.1	10.5	10.0
×	✓	×	31.1	16.7	13.0
✓	×	✓	32.8	18.7	13.9
✓	✓	×	33.9	20.1	14.3
✓	✓	✓	**40.5**	**26.8**	**20.0**

D. Comparison with State-of-the-Art Methods

As shown in Table. I, RSEN performed exceptionally well in all benchmark tests, demonstrating competitive performance. Taking the PredCls task on the VG dataset as an example, RSEN attains 40.5% in mR@100, significantly outperforming Motifs+TDE (29.1%) and Motifs+PCPL (26.1%). These gains demonstrate that RSEN is more effective than traditional causal reasoning or re-weighting schemes. In addition, RSEN effectively addresses the long-tail challenge without compromising the performance on head categories. As illustrated in Fig. 3, the RSEN has significantly improved its average recall rates in the tail categories, while maintaining a stable performance in the head category, demonstrating an outstanding balance in overall scene understanding.

Furthermore, RSEN delivers significant performance gains on the GQA dataset, demonstrating strong dataset-agnostic generalization and not being limited to a specific data distribution.

E. Ablation Studies

Influence of each component: We evaluate the contribution of each RSEN component in Table. II. Starting from the baseline, the HVA and HCI modules independently improve the mR@100 (PredCls) to 25.1% and 31.1% respectively. This indicates that capturing the global context relationships is the foundation for addressing the ambiguity of relationships. When the HVA is combined with the SPR module, the performance

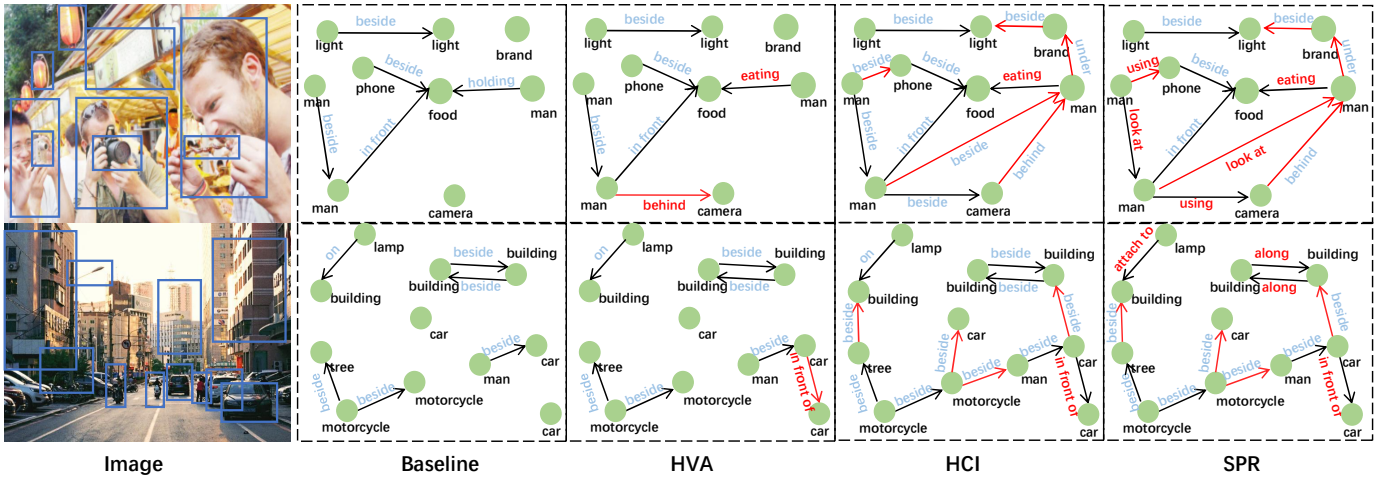

Fig. 4. Progressive enhancement of scene graph generation: effect comparison of module-stacked optimization. Red edges indicate newly detected relationships, while red labels highlight fine-grained predicates refined from coarse-grained ones.

TABLE III
IMPACT OF HYPERPARAMETER N IN HIERARCHICAL CONTEXT INTERACTION MODULE ON MODEL PERFORMANCE (MR@50 AND MR@100 METRICS) ON VG.

N	PredCls		SGCls		SGGen	
	mR@50	mR@100	mR@50	mR@100	mR@50	mR@100
1	31.5	32.9	16.1	17.9	10.8	12.5
2	32.1	33.4	17.7	18.5	11.2	14.9
3	**39.1**	**40.5**	**23.3**	**26.8**	**19.1**	**20.0**
4	38.1	39.2	21.7	23.8	16.9	18.1
5	34.3	37.2	19.6	21.5	15.5	16.3

is further enhanced to 32.8%. This confirms that the semantic prototype refinement (SPR) technique can effectively utilize the semantic prior knowledge of CLIP to optimize the rough features provided by HVA. Finally, our complete RSEN achieved a peak performance of 40.5%, which indicates that integrating these three modules is crucial for capturing the fine semantic depth required for high-quality scene graphs.

As visualized in Fig. 4, the baseline model is often restricted to coarse-grained predicates (e.g., *"on"*). After introducing the HVA module, the precision of the prediction predicates has slightly improved. The integration of the HCI module expands the scope of relationships covered, while the addition of SPR enables the prediction of highly specific predicates (e.g., *"attach to"*). This indicates that RSEN not only improves the indicators, but also enhances the descriptive ability of the scene representation and the explainability of the real world.

Influence of Hyperparameters N: As shown in Table. III, we investigate the impact of the iteration number N on the VG dataset. RSEN achieves its peak performance with three iterations ($N = 3$). Insufficient iterations fail to fully leverage global context, whereas exceeding this threshold introduces feature over-smoothing, thereby diminishing the model's ability to distinguish fine-grained relationships.

V. CONCLUSION

We propose the Relational Semantic Enhancement Network (RSEN) to address the critical challenges of *long-tail distributions* in SGG. By integrating hierarchical context interaction with prototype-guided modeling, RSEN achieves a high degree of structural consistency and semantic granularity. Our research results show that RSEN not only significantly improves the $mR@K$ metric, but also maintains comparable $R@K$ performance to competing algorithms. Future research will focus on real-time optimization to enhance the scalability of the model and to address complex large-scale visual perception tasks.

REFERENCES

[1] A. Zhang, Y. Yao, Q. Chen, W. Ji, Z. Liu, M. Sun, and T.-S. Chua, "Fine-grained scene graph generation with data transfer," in *European conference on computer vision*. Springer, 2022, pp. 409–424.
[2] R. Zellers, M. Yatskar, S. Thomson, and Y. Choi, "Neural motifs: Scene graph parsing with global context," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5831–5840.
[3] K. Tang, H. Zhang, B. Wu, W. Luo, and W. Liu, "Learning to compose dynamic tree structures for visual contexts," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 6619–6628.
[4] D. Xu, Y. Zhu, C. B. Choy, and L. Fei-Fei, "Scene graph generation by iterative message passing," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 5410–5419.
[5] J. Yang, J. Lu, S. Lee, D. Batra, and D. Parikh, "Graph r-cnn for scene graph generation," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 670–685.
[6] J. Shi, H. Zhang, and J. Li, "Explainable and explicit visual reasoning over scene graphs," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 8376–8384.
[7] Y. Cui, M. Jia, T.-Y. Lin, Y. Song, and S. Belongie, "Class-balanced loss based on effective number of samples," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 9268–9277.
[8] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*. PmLR, 2021, pp. 8748–8763.
[9] R. Li, S. Zhang, B. Wan, and X. He, "Bipartite graph network with adaptive message passing for unbiased scene graph generation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 11 109–11 119.

[10] X. Dong, T. Gan, X. Song, J. Wu, Y. Cheng, and L. Nie, "Stacked hybrid-attention and group collaborative learning for unbiased scene graph generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 19 427–19 436.

[11] C. Zheng, X. Lyu, L. Gao, B. Dai, and J. Song, "Prototype-based embedding network for scene graph generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 22 783–22 792.

[12] J. Jeon, K. Kim, K. Yoon, and C. Park, "Semantic diversity-aware prototype-based learning for unbiased scene graph generation," in *European Conference on Computer Vision*. Springer, 2024, pp. 379–395.

[13] K. Tang, Y. Niu, J. Huang, J. Shi, and H. Zhang, "Unbiased scene graph generation from biased training," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 3716–3725.

[14] S. Yan, C. Shen, Z. Jin, J. Huang, R. Jiang, Y. Chen, and X.-S. Hua, "Pcpl: Predicate-correlation perception learning for unbiased scene graph generation," in *Proceedings of the 28th ACM international conference on multimedia*, 2020, pp. 265–273.

[15] J. Yu, Y. Chai, Y. Wang, Y. Hu, and Q. Wu, "Cogtree: Cognition tree loss for unbiased scene graph generation," *arXiv preprint arXiv:2009.07526*, 2020.

[16] L. Liu, S. Sun, S. Zhi, F. Shi, Z. Liu, J. Heikkilä, and Y. Liu, "A causal adjustment module for debiasing scene graph generation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

[17] H. Zhang, Z. Kyaw, S.-F. Chang, and T.-S. Chua, "Visual translation embedding network for visual relation detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 5532–5540.

[18] X. Lin, C. Ding, J. Zeng, and D. Tao, "Gps-net: Graph property sensing network for scene graph generation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 3746–3753.

[19] T. Chen, W. Yu, R. Chen, and L. Lin, "Knowledge-embedded routing network for scene graph generation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 6163–6171.

[20] Z. Liu, J. Wang, H. Chen, Y. Ma, and N. Zheng, "Relation-specific feature augmentation for unbiased scene graph generation," *Pattern Recognition*, vol. 157, p. 110936, 2025.

[21] H. Kim and B. C. Ko, "Semantic scene graph generation based on an edge dual scene graph and message passing neural network," *Image and Vision Computing*, p. 105572, 2025.

[22] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma *et al.*, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *International journal of computer vision*, vol. 123, no. 1, pp. 32–73, 2017.

[23] D. A. Hudson and C. D. Manning, "Gqa: A new dataset for real-world visual reasoning and compositional question answering," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 6700–6709.

[24] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1492–1500.

[25] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, 2015.