HRTR: Hierarchical Reasoning Transformer for Scene Graph Generation

Zengxun Jin[1], Yun Meng[1,*]

[1]*School of Electronics and Control Engineering,Chang'an University, Xi'an, China*
mengyun@chd.edu.cn

***Abstract*—Scene Graph Generation aims to produce a fundamental structured representation for deep visual understanding and downstream reasoning tasks. However, current one-stage methods typically treat categories as flat, isolated labels, ignoring intrinsic semantic correlations. This semantic isolation restricts their generalization, causing failures in zero-shot and long-tailed scenarios. To address this, we propose the Hierarchical Reasoning Transformer (HRTR), an end-to-end framework that integrates structured knowledge priors into the one-stage detection pipeline. Specifically, we construct a Hierarchical Semantic Tree derived from WordNet to explicitly model parent-child dependencies, effectively overcoming the limitations of flat label spaces. Building on this structure, we propose a Semantic-Visual Unification module that dynamically aligns these static priors with visual features, bridging the semantic gap to enable robust structured reasoning. Furthermore, we design a Hierarchical Supervision Loss to enforce structural consistency during optimization, ensuring logical coherence across coarse-to-fine predictions. Extensive experiments on Visual Genome and Open Images V6 demonstrate that HRTR achieves a strong balance between efficiency and performance, with significant gains in zero-shot Recall and mean Recall.**

***Index Terms*—Scene Understanding, Scene Graph, Transformer, Hierarchical Reasoning, Visual Relationship Detection**

I. Introduction

Scene graphs empower intelligent systems to achieve deep semantic understanding by modeling visual scenes as structured representations. Formally defined as directed graphs, they map objects to nodes and relationships to edges, typically represented as $\langle subject, predicate, object \rangle$ triplets. Owing to this rich structural prior, scene graphs have emerged as a foundational component for advancing complex vision tasks, proving instrumental in diverse applications ranging from cross-modal retrieval [1] and grounded captioning [2] to visual reasoning [3] and controllable image synthesis [4]. More recently, they have become crucial for embodied AI [5]. Unlike object detection, SGG models complex interactions among entities. To tackle this, early dominant approaches [6]–[12] adopted a two-stage pipeline: first localizing objects using a detector, and then inferring pairwise relationships with a dedicated reasoning network. While effective, these methods suffer from inherent inefficiencies. They typically require processing $O(n^2)$ pairs for n object proposals, leading to high computational costs. Moreover, the separation of detection and reasoning often prevents the relationship model from correcting errors propagated from the object detector.

To reduce cost, recent work adopts one-stage set prediction, using Hungarian matching over learnable queries for end-to-end training [13]–[16]. By formulating SGG as a direct set prediction problem, these methods employ the Hungarian algorithm [17]–[24] to assign triplets to learnable queries, enabling efficient end-to-end training. However, this efficiency comes at a cost. Conventional one-stage models typically treat object and predicate categories as flat, independent labels, ignoring the intrinsic semantic correlations found in human cognition (e.g., that a "Cat" shares attributes with a "Dog"). This semantic isolation deprives them of the ability to transfer knowledge from frequent to rare classes. Consequently, they tend to overfit to training statistics, failing to generalize to visually plausible but unseen compositions.

To address the efficiency–generalization trade-off, we propose the **H**ierarchical **R**easoning **T**ransforme**R** (HRTR), which extends RelTR by explicitly injecting hierarchical semantic modeling into the Transformer [25]–[28], enabling stronger knowledge transfer from head to tail and zero-shot categories. We evaluate HRTR on Visual Genome [29] and Open Images V6 [30], and the results show that our model maintains competitive performance while achieving high inference efficiency. The main contributions of this work are summarized as follows:

- We construct a **Hierarchical Semantic Tree** derived from WordNet to explicitly model the intrinsic parent-child dependencies among categories. This structure serves as a fundamental prior, overcoming the semantic isolation of flat label spaces used in prior one-stage methods.
- We propose a **Semantic-Visual Unification** module that dynamically aligns these structured knowledge priors with visual features. By fusing static semantic hierarchy with dynamic visual appearance, it effectively bridges the semantic gap, enabling compositional generalization to zero-shot scenarios.
- We design a **Hierarchical Supervision Loss** to enforce **structural consistency** across coarse-to-fine predictions. This objective ensures logical coherence during optimization and regularizes the model.

The source code and data are publicly available at https://github.com/jzx05/HRTR.

*Corresponding author.

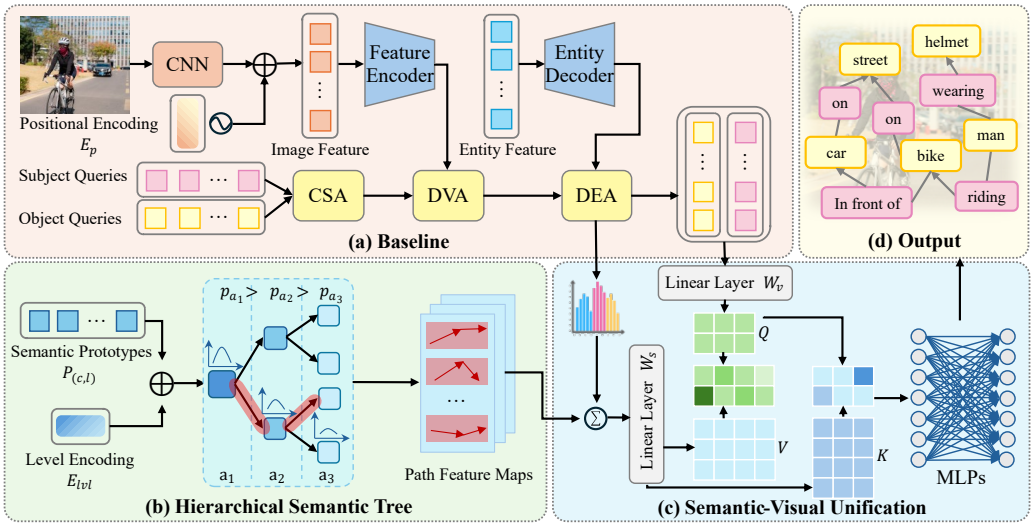

Fig. 1. (a) **Baseline**: The backbone extracts visual features from input images, where CSA, DVA, and DEA modules iteratively refine the subject and object queries. (b) **HST**: Constructs coarse-to-fine semantic paths from ancestors to children ($a_1 \rightarrow a_2 \rightarrow a_3$) using semantic prototypes and level encoding. (c) **SVU**: Fuses the hierarchical path features with visual queries via linear projections and cross-attention mechanism. (d) **Output**: The final scene graph is generated directly, optimized end-to-end by the Hierarchical Supervision Loss to ensure structural consistency.

II. Related Work

A. Scene Graph Generation

Scene graph generation is used to detect the relationships between objects in a scene. Existing scene graph generation methods can be broadly classified into two-stage and one-stage approaches.

The conventional two-stage strategy first employs a strong object detector, such as Faster-RCNN [31], to localize object proposals, followed by a relational reasoning network to infer predicates between pairs of object proposals. Early approaches relied on message-passing mechanisms, utilizing RNNs [32], [33] or Graph Neural Networks [8], [9] to refine context among entity nodes. Recently, Transformer-based architectures [34], [35] have been introduced to capture global dependencies. Although these methods achieve strong performance, they remain computationally expensive due to pairwise relation reasoning over $O(N^2)$ candidates and are subject to error propagation through the frozen detector stage.

To prioritize efficiency, recent works have shifted towards one-stage frameworks. FCSGG [36] encodes objects as centers and relationships as vector fields but struggles with a performance gap. SGTR [14] introduces a decouple-then-assemble strategy. Most notably, RelTR [13] formulates SGG as a direct set prediction task using learnable queries, significantly boosting inference speed. However, these one-stage models typically regard object categories as flat and isolated labels. By neglecting the intrinsic semantic hierarchy (e.g., dog is-an animal), they often fail to capture fine-grained semantic correlations, limiting their effectiveness in zero-shot or long-tailed scenarios.

B. Knowledge-Enhanced Reasoning

SGG remains limited by long-tail relations and weak zero-shot generalization. To mitigate class imbalance, unbiased learning strategies [6], [37] have been widely explored. Beyond data re-balancing, incorporating structured knowledge priors has proven effective. Semantic knowledge [8], [38] and statistical correlations [39] are frequently used to regularize relationship inference. However, most existing knowledge-aware methods are tailored for two-stage methods. Explicit structured hierarchical reasoning remains under-explored in efficient one-stage Transformer frameworks. In this work, we bridge this gap by explicitly embedding structured priors into a one-stage framework to simultaneously enhance efficiency and generalization.

III. Methodology

A. Preliminaries

Our method is built upon the one-stage SGG framework, RelTR [13], which employs a Transformer encoder–decoder architecture to directly predict a fixed set of triplets. The backbone first extracts global visual features $\boldsymbol{Z}$. Subsequently, a dedicated Entity Decoder outputs entity representations $\boldsymbol{Q}_{\text{ent}} \in \mathbb{R}^{N_e \times d}$, where N_e denotes the number of entities and d is the feature dimension. The Triplet Decoder initializes subject queries $\boldsymbol{Q}_{\text{sub}} \in \mathbb{R}^{N_t \times d}$ and object queries $\boldsymbol{Q}_{\text{obj}} \in \mathbb{R}^{N_t \times d}$ to decode relationship information. These queries are refined iteratively through three coupled attention modules:

- Coupled Self-Attention (CSA): Captures contextual interactions among N_t triplets and models the dependencies between all subject and object queries.
- Decoupled Visual Attention (DVA): Updates $\boldsymbol{Q}_{\text{sub}}$ and $\boldsymbol{Q}_{\text{obj}}$ by attending to the visual features $\boldsymbol{Z}$, producing multi-head spatial attention weights $\boldsymbol{V}_{\text{spa}}$ for localization.

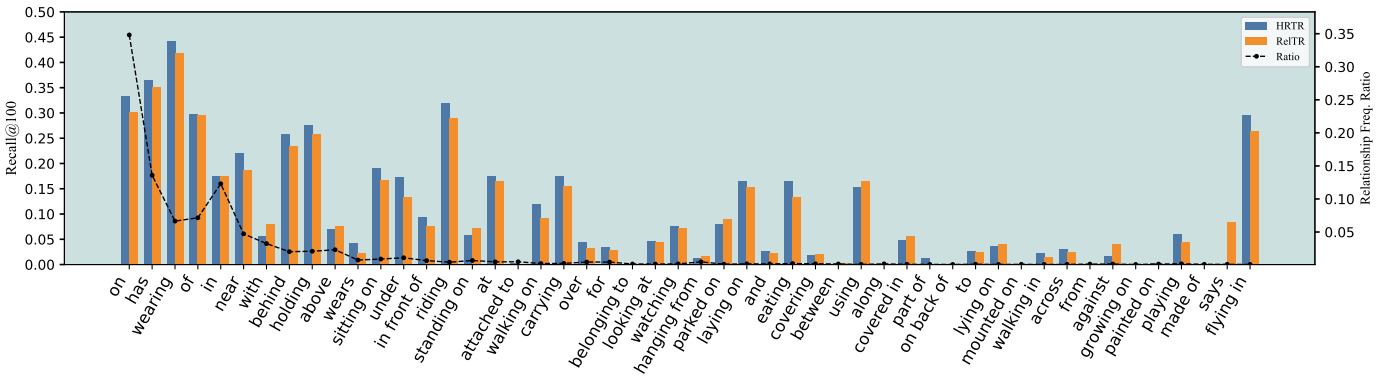

Fig. 2. Recall@100 performance for each relationship category on the SGDet task. The dashed line with dots denotes the frequency of each relationship category in the dataset, while the two colored bars present the comparative performance of HRTR and RelTR.

- Decoupled Entity Attention (DEA): Refines the localization and classification features of $\boldsymbol{Q}_{\text{sub}}$ and $\boldsymbol{Q}_{\text{obj}}$ by attending to entity representations $\boldsymbol{Q}_{\text{ent}}$ from the Entity Decoder.

The predicted triplet is generated by projecting the $\boldsymbol{Q}_{\text{sub}}$ and $\boldsymbol{Q}_{\text{obj}}$ into class and box coordinates. The spatial attention weights are fused with the queries to form visual features $\boldsymbol{V}_{\text{spa}}$, which are then fed into MLPs for predicate classification.

During training, RelTR optimizes a joint objective comprising entity and triplet detection losses. Ground truths are padded to fixed sizes (N_e and N_t) with the background token $\varnothing$, and assigned to predictions via bipartite matching using the Hungarian algorithm [17]. The entity loss $\mathcal{L}_{\text{entity}}$ aggregates classification and box regression costs:

$$\mathcal{L}_{\text{entity}} = \sum_{i=1}^{N_e} \left[\mathcal{L}_{\text{cls}}(i) + \mathbf{1}_{\{c_i \neq \varnothing\}} \mathcal{L}_{\text{box}}(i) \right], \tag{1}$$

where $\mathcal{L}_{\text{cls}}$ denotes the cross-entropy loss, $\mathcal{L}_{\text{box}}$ combines ℓ_1 and GIoU losses [40], and $1_{\{\cdot\}}$ is the indicator function. The triplet loss $\mathcal{L}_{\text{triplet}}$ aggregates costs for the **s**ubject (s), **o**bject (o), and **p**redicate (p). It sums the classification loss for all components and the localization loss for triplet entities:

$$\mathcal{L}_{\text{triplet}} = \sum_{i=1}^{N_t} \Big[\sum_{k \in \{s,o,p\}} \mathcal{L}^k_{\text{cls}}(i) + \mathbf{1}_{\{c_i \neq \varnothing\}} \sum_{k \in \{s,o\}} \mathcal{L}^k_{\text{box}}(i) \Big], \tag{2}$$

The total loss function is the sum of the two:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{entity}} + \mathcal{L}_{\text{triplet}}. \tag{3}$$

B. Model Architecture

Figure 1 illustrates the overall architecture of our proposed HRTR. Building upon the RelTR [13] baseline, we integrate a semantic hierarchy enhancement mechanism to bridge the semantic gap and facilitate structured reasoning. Specifically, we introduce three key components: a Hierarchical Semantic Tree (HST), a Semantic-Visual Unification (SVU), and a Hierarchical Supervision Loss (HSL).

Hierarchical Semantic Tree (HST): To explicitly capture the rich semantic relationships between subjects and objects, we construct HST based on the WordNet knowledge base. Let $|\mathcal{C}_e|$ represent the complete set of entity classes. Instead of treating each class as an isolated point, we define the representation of any category $c \in |\mathcal{C}_e|$ as an ordered semantic path of length L:

$$\mathcal{A}(c) = \langle a_1(c), a_2(c), \ldots, a_L(c) \rangle, \tag{4}$$

where $a_L(c) = c$ denotes the specific class, and $a_1(c)$ corresponds to the root abstraction. For instance, a typical semantic path is *Animal* $\rightarrow$ *Mammal* $\rightarrow$ *Dog*. The hierarchy follows parent-to-child connections defined as $a_{\ell-1}(c) \rightarrow a_\ell(c)$ for $\ell = 2, \ldots, L$. By aggregating features along this path, the hierarchy promotes structural knowledge sharing and enhances the generalization capability for zero-shot scenarios.

To encode the hierarchy into a continuous feature space, we introduce a learnable semantic prototype bank $\mathbf{P} \in \mathbb{R}^{|\mathcal{C}_e| \times L \times d}$ to capture category-specific semantics. Additionally, we employ shared level embeddings $\mathbf{E}_{\text{lvl}} \in \mathbb{R}^{L \times d}$ to explicitly differentiate semantic abstractions at varying depths. For the semantic path of any class c, the hierarchical prototype sequence $\mathbf{H}(c) \in \mathbb{R}^{L \times d}$ is obtained by:

$$\mathbf{H}(c) = \text{LN}\big([\mathbf{P}_{(c,\ell)} + \mathbf{E}_{\text{lvl}}(\ell)]_{\ell=1}^{L} \big), \tag{5}$$

where $\text{LN}(\cdot)$ denotes Layer Normalization, used to stabilize the feature distribution across different semantic levels.

Semantic-Visual Unification (SVU): The SVU module is designed to dynamically bridge the gap between flat visual features and structured knowledge. We utilize the refined classification scores $\mathbf{p} \in \mathbb{R}^{|\mathcal{C}_e|}$ from the DEA module to construct a subject-specific semantic representation. Instead of relying on a hard label, we compute a weighted aggregation of hierarchical prototypes:

$$\mathbf{S}_{\text{sub}} = \sum_{c=1}^{|\mathcal{C}_e|} p_{\text{sub}}(c)\, \mathbf{H}(c), \tag{6}$$

where $p_{\text{sub}}(c)$ is the probability of the subject belonging to class c, and $\mathbf{S}_{\text{sub}} \in \mathbb{R}^{L \times d}$ encapsulates the multi-level semantic context. The object representation $\mathbf{S}_{\text{obj}}$ is computed analogously.

TABLE I
COMPARISON OF RESULTS OBTAINED FROM EXPERIMENTS ON VISUAL GENOME.

Type	Method	SGDET								FPS↑	#params(M) ↓
		R@20	R@50	R@100	mR@20	mR@50	mR@100	zR@50	zR@100		
two-stage (unbiased)	Motifs-TDE [6]	12.4	16.9	20.3	5.8	8.2	9.8	2.3	2.9	-	270.3
	VCTree-TDE [6]	14.3	19.6	23.3	6.3	8.6	10.3	**2.6**	3.2	0.8	361.3
	VCTree-TDE(EBM) [41]	14.7	20.6	-	**7.1**	9.7	11.6	1.6	2.7	-	372.5
two-stage (w/o unbiased)	Motifs [8]	21.4	27.2	30.3	4.2	5.7	6.6	0.1	0.2	6.6	270.4
	KERN [39]	22.3	27.1	29.8	-	6.4	7.3	-	-	4.6	405.2
	VCTree [9]	22.0	27.9	31.3	5.2	6.9	8.0	0.2	0.7	0.8	361.4
one-stage (w/o unbiased)	FCSGG [36]	16.1	21.3	25.1	2.7	3.6	4.2	1.0	1.4	8.4	87.1
	HOTR [42]	-	23.5	27.7	-	9.4	12.0	-	-	-	-
	RelTR [13]	21.2	27.5	29.8	6.8	10.2	12.6	1.8	2.4	**16.1**	63.7
	UniQ [43]	**25.2**	**30.5**	33.2	6.3	8.5	9.6	-	-	-	66.8
	EGTR [44]	23.5	30.2	**34.3**	5.5	7.9	10.1	-	-	14.7	**42.5**
	HRTR(**ours**)	21.6	28.1	30.5	7.0	**10.6**	**13.1**	**2.6**	**3.3**	16.0	69.4

To fuse these modalities, we employ a cross-attention mechanism where the visual features query the semantic hierarchy. We project the visual query $\mathbf{Q}_{\text{sub}}$ and semantic memory $\mathbf{S}_{\text{sub}}$ into a shared latent space via linear projections $\mathbf{W}_Q$ and $\mathbf{W}_{K,V}$. The fused feature $\mathbf{F}_{\text{sub}}$ is obtained by:

$$\mathbf{F}_{\text{sub}} = \text{Attention}\left(\mathbf{Q}_{\text{sub}}\mathbf{W}_Q, \mathbf{S}_{\text{sub}}\mathbf{W}_K, \mathbf{S}_{\text{sub}}\mathbf{W}_V\right). \quad (7)$$

Finally, the predicate distribution is predicted by concatenating the fused subject/object features with spatial union features $\mathbf{V}_{\text{spa}}$:

$$\hat{p}_{\text{prd}} = \text{Softmax}\left(\text{MLP}\left(\mathbf{F}_{\text{sub}}, \mathbf{F}_{\text{obj}}, \mathbf{V}_{\text{spa}}\right)\right). \quad (8)$$

Hierarchical Supervision Loss (HSL): To enforce structural consistency, we introduce HSL. Let $\mathcal{V}$ denote the set of all semantic nodes. For a ground-truth leaf class c^*, its semantic ancestry path is defined as the ordered sequence $\mathcal{A}(c^*) = \langle a_1(c^*), \ldots, a_L(c^*)\rangle$.

We define a binary ancestry mapping matrix $\mathbf{M} \in \{0,1\}^{|\mathcal{C}_e|\times|\mathcal{V}|}$, where $M_{i,j} = 1$ if leaf c_i is a descendant of node v_j. The probabilities of all hierarchical nodes $\mathbf{p}^{\text{node}} \in \mathbb{R}^{|\mathcal{V}|}$ are inferred by aggregating leaf predictions $\mathbf{p}$:

$$\mathbf{p}^{\text{node}} = \mathbf{p}^\top \mathbf{M}. \quad (9)$$

Crucially, this aggregation naturally enforces probability monotonicity ($p_{\text{parent}} \geq p_{\text{child}}$) and bounds all node probabilities within $[0,1]$, ensuring theoretical consistency without external constraints.

During training, we aim to maximize the likelihood of every node along the ground-truth path. The HSL is computed as the average negative log-likelihood over the ancestor chain:

$$\mathcal{L}_{\text{hier}} = -\frac{1}{|\mathcal{A}(c^*)|}\sum_{v\in\mathcal{A}(c^*)} \log \mathbf{p}^{\text{node}}(v). \quad (10)$$

We note that HSL introduces negligible computational overhead, as the aggregation is a single matrix multiplication. The final total loss is:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{entity}} + \mathcal{L}_{\text{triplet}} + \lambda_{\text{hier}}\mathcal{L}_{\text{hier}}, \quad (11)$$

where λ_{hier} is a hyperparameter balancing the structural constraint.

IV. EXPERIMENTS

A. Datasets and Evaluation Settings

Visual Genome: We evaluate our model on the Visual Genome dataset [29], which comprises 108k images annotated with 150 object categories and 50 relationship categories. Following standard protocols [32], we split the data into 70% for training and 30% for testing, with 5k training images reserved for validation. The evaluation encompasses three standard settings: (1) Predicate Classification (**PredCls**): predicting relationships given ground-truth object categories and bounding boxes; (2) Scene Graph Classification (**SGCls**): predicting object categories and relationships given ground-truth bounding boxes; (3) Scene Graph Detection (**SGDet**): detecting both objects and relationships directly from raw images. Given the end-to-end nature of our method, we primarily focus on the most challenging **SGDet** task. Performance is reported using Recall@K (R@K), mean Recall@K (mR@K), and zero-shot Recall@K (zR@K).

Open Images V6: We conduct experiments on the large-scale Open Images V6 [30] consisting of 126k training images, 5.3k test images, and 1.8k validation images. It involves 288 entity categories and 30 predicate categories. We adopt the standard evaluation metrics used in the Open Images Challenge. Recall@50, weighted mean average precision (AP) of relationship detection wmAP_{rel}, and phrase detection wmAP_{phr}. The final score is computed as:

$$\text{score}_{\text{wtd}} = 0.2\times R@50 + 0.4\times \text{wmAP}_{\text{rel}} + 0.4\times \text{wmAP}_{\text{phr}}.$$

B. Implementation Details

We implement HRTR in PyTorch using ResNet-50 as the backbone. Training is conducted on 4 RTX 4090 GPUs for 150 epochs using AdamW [45] (batch size 16, weight decay 10^{-4}). Learning rates are initialized at 10^{-5} for the backbone and 10^{-4} for the Transformer, decaying by 0.1 at epoch 100. To ensure stability, SVU and HSL are activated after epoch 110, adopting a curriculum strategy to prioritize visual representation learning. The Transformer employs a 6-layer encoder-decoder ($d = 256$, 8 heads) with $N_e = 100$, $N_t = 200$, hierarchy depth $L = 4$, and $\lambda_{\text{hier}} = 0.5$.

C. Quantitative Results and Comparison

Visual Genome: We benchmark HRTR against unbiased two-stage, general one-stage, and general two-stage methods on Visual Genome [29] utilizing R@K, mR@K, and zR@K metrics, as summarized in Table I. While unbiased two-stage methods show a marginal gain in mR@20 (0.1% higher than HRTR), they suffer a significant drop in standard R@K. In the one-stage approaches, although UniQ [43] achieves the best R@20/50 and EGTR [44] leads at R@100, they fall behind HRTR in terms of mR@K. Despite its lightweight design (69.4M parameters, 16.0 FPS), HRTR delivers consistent improvements across all metrics, with competitive R@K and the strongest mR@K and zR@K among one-stage models.

The detailed breakdown in Fig. 2 further validates this trend. HRTR consistently outperforms the RelTR baseline across most predicate categories. This advantage is evident not only in high-frequency predicates (e.g., ***wearing***, ***near***) but becomes even more pronounced for tail categories. This demonstrates that HRTR effectively mitigates semantic bias, yielding robust and balanced predictions for complex scene graphs.

Open Images V6: We train HRTR on Open Images V6 dataset [30] and compare it with other methods, as shown in Table II. HRTR demonstrates robust and superior performance. Although R@50 of HRTR is 2.5% lower than the two-stage method BGNN [46], HRTR has the higher wmAP_{rel} (2.7% higher than BGNN) and wmAP_{phr} (4.2% higher than BGNN). While wmAP_{phr} of HRTR is 0.7% lower than BCTR [47], its R@50 is 3.8% higher. The final weighted score of HRTR is 0.7% higher than the best model BCTR. The inference speed of HRTR is 16.2 FPS, which is only 0.1 FPS lower than RelTR [13]. However, its $\text{score}_{\text{wtd}}$ is 1.3% higher.

TABLE II
COMPARISON OF RESULTS ON OPEN IMAGES V6. † DENOTES OUR RE-IMPLEMENTATION FOLLOWING THE SETTING IN THE PAPER.

Method	R@50	$wmAP_{rel}$	$wmAP_{phr}$	$score_{wtd}$	FPS↑
Motifs [8]	71.6	29.9	31.6	38.9	7.4
RelDN [48]	72.8	29.9	30.4	38.7	5.3
GPS-Net [49]	74.7	32.8	33.9	41.6	-
BGNN [46]	**74.9**	33.5	34.1	41.7	2.9
AS-Net [50]	55.3	15.9	27.5	32.4	-
SGTR† [14]	71.0	35.5	38.8	42.4	-
RelTR [13]	71.7	34.2	37.5	43.0	**16.3**
BCTR [47]	68.6	36.0	**39.0**	43.7	-
HRTR(**ours**)	72.4	**36.2**	38.3	**44.3**	16.2

D. Ablation Studies

To verify the effectiveness of each component, we evaluate HST, SVU, and HSL in isolation. As shown in Table III, the baseline exhibits limited performance on SGDet, achieving only 12.6 mR@100 and 2.4 zR@100. Introducing modules individually reveals distinct benefits. HST delivers the largest boost to long-tail and zero-shot capabilities, raising zR@100 to 3.1 and mR@100 to 12.8. In contrast, using SVU alone leads to scores lower than the baseline, likely due to optimization instability without structured knowledge priors. HSL primarily enhances detection coverage (R@50/100), having a limited impact on tail and zero-shot results.

Combining modules demonstrates strong complementarity. Notably, the HST and SVU combination achieves the highest mR@50 (10.7) among dual settings while maintaining competitive zR@100, validating the synergy between semantic structure and visual alignment.

Activating all three modules yields the peak performance, reaching 30.5 R@100, 13.1 mR@100, and 3.3 zR@100. This confirms that the integration of HST, SVU, and HSL is essential for robust relation detection, simultaneously improving accuracy, long-tail recognition, and zero-shot generalization.

TABLE III
COMPARISON OF RESULTS OBTAINED FROM ABLATION STUDY.

Ablation Setting			SGDET				
HST	SVU	HSL	R@50	R@100	mR@50	mR@100	zR@100
✗	✗	✗	27.5	29.8	10.2	12.6	2.4
✓	✗	✗	27.4	29.6	10.2	12.8	3.1
✗	✓	✗	27.5	29.7	10.0	12.0	2.4
✗	✗	✓	28.0	**30.5**	9.1	11.8	2.7
✓	✓	✗	27.7	30.0	**10.7**	13.0	3.0
✓	✗	✓	28.0	30.3	9.9	12.7	2.1
✗	✓	✓	27.5	29.6	10.1	11.9	2.5
✓	✓	✓	**28.1**	**30.5**	10.6	**13.1**	**3.3**

The weight λ_{hier} plays a role in balancing long-tailed recognition and zero-shot generalization. As illustrated in Fig. 3, mR@100 peaks at $\lambda_{\text{hier}} = 0.5$, suggesting that moderate supervision is optimal for preserving fine-grained discrimination in tail categories. Conversely, zR@100 reaches its maximum at 0.75, indicating that zero-shot inference relies more heavily on strong structural guidance to transfer knowledge. Since performance on both metrics degrades at $\lambda_{\text{hier}} = 1.0$, we identify $[0.5, 0.75]$ as the optimal interval, effectively striking a balance between hierarchical constraints and visual representation learning.

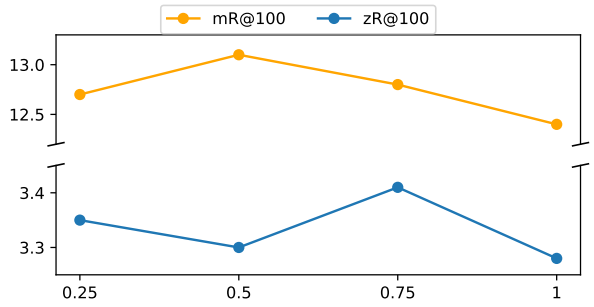

Fig. 3. mRecall@100 and zRecall@100 curves of HRTR on the SGDet task under different values of λ_{hier}.

V. CONCLUSION

We have introduced HRTR, an efficient one-stage SGG framework that leverages structured knowledge priors to enhance semantic reasoning. Through the synergy of hierarchical prototype learning and structure-aware supervision, HRTR achieves a strong balance between standard recall and long-tail generalization. Experiments on Visual Genome and Open

Images V6 validate that our method outperforms existing one-stage baselines in mean Recall and zero-shot scenarios without compromising inference speed. This work highlights the critical role of structured semantic constraints in improving the robustness of end-to-end scene graph generation.

ACKNOWLEDGMENT

This work was partially supported by the National Key R&D Program of China (No. 2024YFF0908100) and the Key R&D Program of Shaanxi (No. 2024CY2-GJHX-59).

REFERENCES

[1] Y. Miao, F. Engelmann, O. Vysotska, F. Tombari, M. Pollefeys, and D. B. Baráth, "Scenegraphloc: Cross-modal coarse visual localization on 3d scene graphs," in *ECCV*. Springer, 2024.

[2] K. Bae, J. Kim, S. Lee, S. Lee, G. Lee, and J. Choi, "Mash-vlm: Mitigating action-scene hallucination in video-llms through disentangled spatial-temporal representations," in *CVPR*, 2025.

[3] J. Gao, R. Pi, J. Zhang, J. Ye, W. Zhong, Y. Wang, L. Hong, J. Han, H. Xu, Z. Li *et al.*, "G-llava: Solving geometric problem with multi-modal large language model," *arXiv preprint arXiv:2312.11370*, 2023.

[4] A. Farshad, Y. Yeganeh, Y. Chi, C. Shen, B. Ommer, and N. Navab, "Scenegenie: Scene graph guided diffusion models for image synthesis," in *ICCV*, 2023.

[5] Q. Gu, A. Kuwajerwala, S. Morin, K. M. Jatavallabhula, B. Sen, A. Agarwal, C. Rivera, W. Paul, K. Ellis, R. Chellappa *et al.*, "Conceptgraphs: Open-vocabulary 3d scene graphs for perception and planning," in *ICRA*. IEEE, 2024.

[6] K. Tang, Y. Niu, J. Huang, J. Shi, and H. Zhang, "Unbiased scene graph generation from biased training," in *CVPR*, 2020.

[7] A. Desai, T.-Y. Wu, S. Tripathi, and N. Vasconcelos, "Learning of visual relations: The devil is in the tails," in *ICCV*, 2021.

[8] R. Zellers, M. Yatskar, S. Thomson, and Y. Choi, "Neural motifs: Scene graph parsing with global context," in *CVPR*, 2018.

[9] K. Tang, H. Zhang, B. Wu, W. Luo, and W. Liu, "Learning to compose dynamic tree structures for visual contexts," in *CVPR*, 2019.

[10] Y. Li, C. Yang, H. Zeng, Z. Dong, Z. An, Y. Xu, Y. Tian, and H. Wu, "Frequency-aligned knowledge distillation for lightweight spatiotemporal forecasting," in *ICCV*, 2025.

[11] Y. Li, K. Li, X. Yin, Z. Yang, Z. Dong, Z. Yao, H. Xu, Y. Tian, and Y. Lu, "Sepprune: Structured pruning for efficient deep speech separation," in *AAAI*, 2026.

[12] Y. Li, S. Meng, C. Yang, W. Feng, J. Liu, Z. An, Y. Wang, and Y. Tian, "A comprehensive survey of interaction techniques in 3d scene generation," *Authorea Preprints*, 2026.

[13] Y. Cong, M. Y. Yang, and B. Rosenhahn, "Reltr: Relation transformer for scene graph generation," *TPAMI*, 2023.

[14] R. Li, S. Zhang, and X. He, "Sgtr: End-to-end scene graph generation with transformer," in *CVPR*, 2022.

[15] Y. Li, Z. Zhou, Z. Peng, J. Dong, H. You, R. Yan, S. Wen, Y. Tian, and T. Huang, "A preference-driven methodology for efficient code generation," *IEEE Transactions on Artificial Intelligence*, 2025.

[16] Y. Li, H. Zeng, F. Zhang, C. Yang, Y. Li, and W. Ding, "Efficient Medical Image Segmentation via Reinforcement Learning-Driven K-Space Sampling," *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2025.

[17] R. Stewart, M. Andriluka, and A. Y. Ng, "End-to-end people detection in crowded scenes," in *CVPR*, 2016.

[18] C. Xiao, J. Dou, Z. Lin, Z. Ke, and L. Hou, "From points to coalitions: Hierarchical contrastive shapley values for prioritizing data samples," in *AAAI*, 2026.

[19] C. Xiao, T. Xu, Y. Jiang, H. Gao, Y. Wu *et al.*, "Reversible primitive–composition alignment for continual vision–language learning," in *ICLR*, 2026.

[20] J. Zhang, C. Zhao, C. Xiao, R. Duan, W. Mo, H. Gao, and W. Wang, "Pi-cca: Prompt-invariant cca certificates for replay-free continual multimodal learning," in *ICLR*, 2026.

[21] X. Chen, C. Xiao, and Y. Liu, "Confusion-resistant federated learning via diffusion-based data harmonization on non-iid data," in *NeurIPS*, 2024.

[22] Y. Li, K. Ding, and Others, "Ddtime: Dataset distillation with spectral alignment and information bottleneck for time-series forecasting," *arXiv preprint arXiv:2511.16715*, 2025.

[23] Y. Li, K. Ding, C. Yang, S.-Y. Chen, and Y. Tian, "Distilling time series foundation models for efficient forecasting," *arXiv preprint arXiv:2601.12785*, 2026.

[24] J. Liu, Y. Li, S. Wen, Z. Zeng, and T. Huang, "Hit-rag: Learning to reason with long contexts via preference alignment," *arXiv preprint arXiv:2603.07023*, 2026.

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *NeurIPS*, 2017.

[26] Z. Xu, X. Zhang, R. Li, Z. Tang, Q. Huang, and J. Zhang, "Fakeshield: Explainable image forgery detection and localization via multi-modal large language models," in *ICLR*, 2025.

[27] Z. Xu, X. Zhang, X. Zhou, and J. Zhang, "Avatarshield: Visual reinforcement learning for human-centric video forgery detection," *arXiv preprint arXiv:2505.15173*, 2025.

[28] Q. Huang, Z. Xu, X. Zhang, and J. Zhang, "Unishield: An adaptive multi-agent framework for unified forgery image detection and localization," *arXiv preprint arXiv:2510.03161*, 2025.

[29] 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," *IJCV*, 2017.

[30] A. Kuznetsova, H. Rom *et al.*, "The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale," *IJCV*, 2020.

[31] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *NeurIPS*, 2015.

[32] D. Xu, Y. Zhu, C. B. Choy, and L. Fei-Fei, "Scene graph generation by iterative message passing," in *CVPR*, 2017.

[33] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "Lstm: A search space odyssey," *TNNLS*, 2016.

[34] Y. Lu, H. Rai, J. Chang, B. Knyazev, G. Yu, S. Shekhar, G. W. Taylor, and M. Volkovs, "Context-aware scene graph generation with seq2seq transformers," in *ICCV*, 2021.

[35] N. Dhingra, F. Ritter, and A. Kunz, "Bgt-net: Bidirectional gru transformer network for scene graph generation," in *CVPR*, 2021.

[36] H. Liu, N. Yan, M. Mortazavi, and B. Bhanu, "Fully convolutional scene graph generation," in *CVPR*, 2021.

[37] 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.

[38] H. Zhang, Z. Kyaw, S.-F. Chang, and T.-S. Chua, "Visual translation embedding network for visual relation detection," in *CVPR*, 2017.

[39] T. Chen, W. Yu, R. Chen, and L. Lin, "Knowledge-embedded routing network for scene graph generation," in *CVPR*, 2019.

[40] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," in *CVPR*, 2019.

[41] M. Suhail, A. Mittal, B. Siddiquie, C. Broaddus, J. Eledath, G. Medioni, and L. Sigal, "Energy-based learning for scene graph generation," in *CVPR*, 2021.

[42] B. Kim, J. Lee, J. Kang, E.-S. Kim, and H. J. Kim, "Hotr: End-to-end human-object interaction detection with transformers," in *CVPR*, 2021.

[43] X. Liao, W. Wei, D. Chen, and Y. Fu, "Uniq: Unified decoder with task-specific queries for efficient scene graph generation," in *ACM MM*, 2024.

[44] J. Im, J. Nam, N. Park, H. Lee, and S. Park, "Egtr: Extracting graph from transformer for scene graph generation," in *CVPR*, 2024.

[45] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[46] R. Li, S. Zhang, B. Wan, and X. He, "Bipartite graph network with adaptive message passing for unbiased scene graph generation," in *CVPR*, 2021.

[47] P. Hao, W. Wang, and Others, "Bctr: bidirectional conditioning transformer for scene graph generation," *Information Fusion*, 2025.

[48] J. Zhang, K. J. Shih, A. Elgammal, A. Tao, and B. Catanzaro, "Graphical contrastive losses for scene graph parsing," in *CVPR*, 2019.

[49] X. Lin, C. Ding, J. Zeng, and D. Tao, "Gps-net: Graph property sensing network for scene graph generation," in *CVPR*, 2020.

[50] M. Chen, Y. Liao, S. Liu, Z. Chen, F. Wang, and C. Qian, "Reformulating hoi detection as adaptive set prediction," in *CVPR*, 2021.