International Journal of Computer Vision (2023) 131:2489–2515
https://doi.org/10.1007/s11263-023-01817-7

MANUSCRIPT

Importance First: Generating Scene Graph of Human Interest

Wenbin Wang[1,2] · **Ruiping Wang**[1,2] · **Shiguang Shan**[1,2] · **Xilin Chen**[1,2]

Received: 24 August 2022 / Accepted: 9 May 2023 / Published online: 9 June 2023

Abstract
Scene graph aims to faithfully reveal humans' perception of image content. When humans look at a scene, they usually focus on their interested parts in a special priority. This innate habit indicates a hierarchical preference about human perception. Therefore, we argue to generate the Scene Graph of Interest which should be hierarchically constructed, so that the important primary content is firstly presented while the secondary one is presented on demand. To achieve this goal, we propose the Tree–Guided Importance Ranking (TGIR) model. We represent the scene with a hierarchical structure by firstly detecting objects in the scene and organizing them into a Hierarchical Entity Tree (HET) according to their spatial scale, considering that larger objects are more likely to be noticed instantly. After that, the scene graph is generated guided by structural information of HET which is modeled by the elaborately designed Hierarchical Contextual Propagation (HCP) module. To further highlight the key relationship in the scene graph, all relationships are re-ranked through additionally estimating their importance by the Relationship Ranking Module (RRM). To train RRM, the most direct way is to collect the key relationship annotation, which is the so-called Direct Supervision scheme. As collecting annotation may be cumbersome, we further utilize two intuitive and effective cues, visual saliency and spatial scale, and treat them as Approximate Supervision, according to the findings that these cues are positively correlated with relationship importance. With these readily available cues, the RRM is still able to estimate the importance even without key relationship annotation. Experiments indicate that our method not only achieves state-of-the-art performances on scene graph generation, but also is expert in mining image-specific relationships which play a great role in serving subsequent tasks such as image captioning and cross-modal retrieval.

Keywords Key relationship · Hierarchical entity tree · Hierarchical contextual propagation · Relationship ranking · Spatial scale · Visual saliency

1 Introduction

In an effort to thoroughly understand a scene, the scene graph consisting of objects as nodes and relationships as edges has been on the way to bridge the gap between low-level recognition and high-level cognition, and contributes to tasks like cross-modal retrieval (Johnson et al., 2015; Wang et al., 2020), image captioning (Chen et al., 2020; Gu et al., 2019; Li & Jiang, 2019; Nguyen et al., 2021; Xu et al., 2019; Yang et al., 2019; Yao et al., 2018; Zhong et al., 2020), visual question answering (Antol et al., 2015; Tang et al., 2019), visual reasoning (Shi et al., 2019), image generation (Gu et al., 2019; Herzig et al., 2020; Johnson et al., 2018) and image editing (Dhamo et al., 2020). While previous works (Guo et al., 2021; Li et al., 2021, 2017; Suhail et al., 2021; Tang et al., 2020, 2019; Wang et al., 2021, 2019; Zareian et al., 2020a; Zellers et al., 2018) have pushed this area forward, the generated scene graph may be still far from perfect. A scene graph is not just for being admired, but is a type of intermediate representations for supporting applications. To this end, the scene graph is expected to at least sketch the major content, i.e., gist, in the scene, which is generally of human interest. However, most generated scene graphs fail

✉ Ruiping Wang
wangruiping@ict.ac.cn

Wenbin Wang
wenbin.wang@vipl.ict.ac.cn

Shiguang Shan
sgshan@ict.ac.cn

Xilin Chen
xlchen@ict.ac.cn

[1] Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences (CAS), Beijing 100190, China

[2] University of Chinese Academy of Sciences, Beijing 100049, China

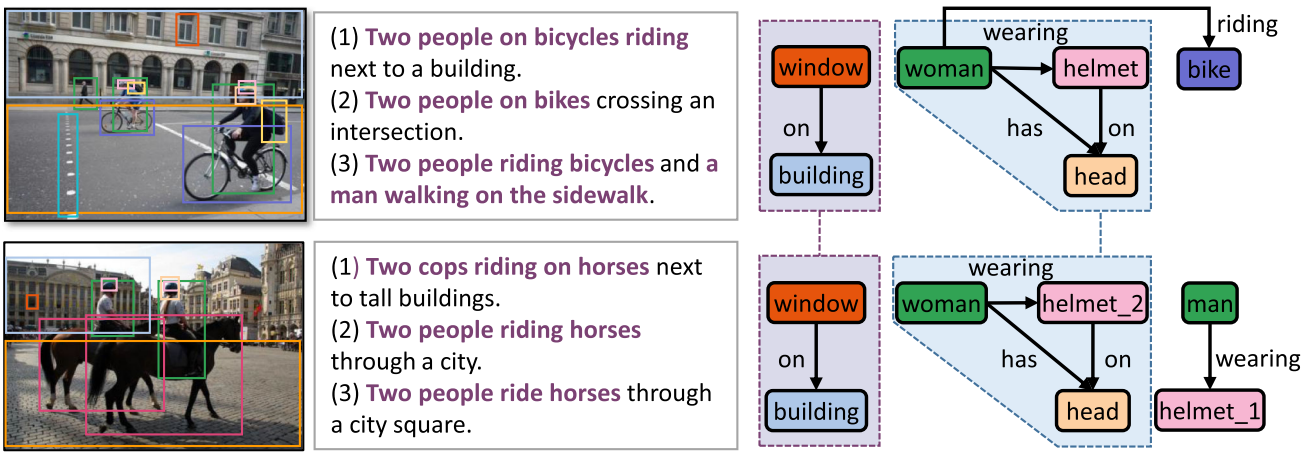


Fig. 1 Scene graphs (right column) of two images with different major events generated by existing methods share similar structures (shown in dashed regions), i.e., existing methods are deficient in mining the major image-specific relationships which are usually embedded in image captions (middle column)

in this field in practice because current methods blindly pursue content integrity, bringing the side effect that the gist is overwhelmed by a large amount of trivial ones (Li & Jiang, 2019).

Let's study the quality of scene graph generated by one of the mainstream methods, Motif (Zellers et al., 2018). In Fig. 1, two scene graphs shown with top-5 relationships for two images are mostly the same although the major events in these two images are quite different. In other words, existing methods are deficient in mining the major image-specific relationships which are usually embedded in image caption (colored parts of the image captions in Fig. 1), but biased towards self-evident or trivial ones (e.g., ⟨*woman*, *has*, *head*⟩ can be obtained from commonsense without observing the image, and ⟨*window*, *on*, *building*⟩ does not deserve much attention compared with other contents in both two images). This shortage makes current scene graph filled with a large body of relationships that are hardly concerned by humans.

Therefore, an urgently needed characteristic of a scene graph is to assess the relationship importance and prioritize the key relationships which form major content that humans intend to preferentially convey. A few previous works, e.g., Graph R-CNN (Yang et al., 2018) and AVR (Lv et al., 2020), consider that annotated relationships are important, and train a light-weight binary classifier to filter unannotated pairs. However, because of the universal phenomenon of long-tailed distribution of relationships in mainstream scene graph datasets such as Visual Genome (Krishna et al., 2017), the annotated pairs are not necessarily important, but usually trivial instead (see the statistical analysis in Sect. 4.1 for further details).

Any pair of objects in a scene can be considered relevant, at least in terms of their spatial configurations. Faced with such a massive amount of relationships, how do humans choose them to describe the images? Given the picture (ii) in Fig. 2a which is a zoom-in sub-region of picture (i), humans will describe it with ⟨*woman*, *riding*, *bike*⟩, since *woman* and *bike* belong to the same perceptive level and their interaction forms the major event in (ii). When it comes to picture (iii), the answers would be ⟨*woman*, *wearing*, *helmet*⟩ and ⟨*bag*, *on*, *woman*⟩, where *helmet* and *bag* are finer details of *woman* and belong to an inferior perceptive level. It suggests that there naturally exists a hierarchical structure about humans' perception preference, as shown in bottom part of Fig. 2a. If we attach the scene graph to this hierarchical structure, as shown in Fig. 2b, the relationships will be naturally presented in a top-down manner and the top relationships are especially of human interest.

Inspired by these observations, we argue that a desirable scene graph should be configured with a hierarchical structure, resulting in the **Scene Graph of Interest (SGoI)**. In SGoI, key relationships are grounded to the top levels of the hierarchical structure, while the secondary or trivial relationships are grounded to the bottom levels. To achieve this goal, we develop the Tree-Guided Importance Ranking (TGIR) model. We firstly construct the Hierarchical Entity Tree (HET) comprising of objects (as nodes), each of which can be decomposed into a set of finer ones. Different tree levels stand for the perception priority of the objects. The structural information of HET is modeled by the elaborately designed Hierarchical Contextual Propagation (HCP) module. HCP takes the object features from the object detector as input, and passes messages among sibling nodes (transversal direction) and nodes on each root-to-leaf path (longitudinal direction) in HET to enhance object features. In this way, each object obtains its own perception level information and contextual information from those closely related objects, which will benefit the importance ranking of relationships

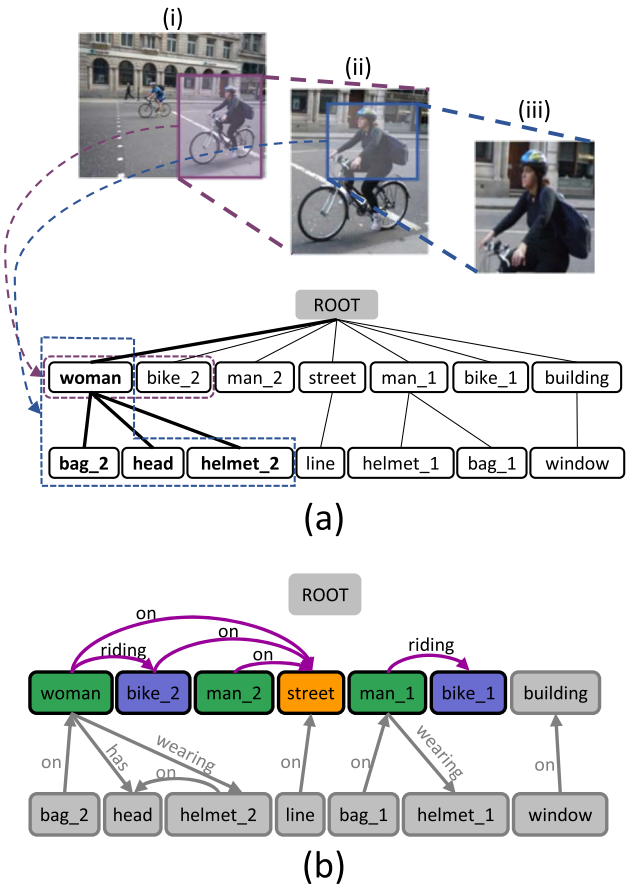


Fig. 2 The hierarchical structure about humans' perception preference is shown in the bottom part of **a** and pictures (i)∼(iii) illustrate a sub-hierarchy. Our generated scene graph in **b** is attached to the hierarchical structure and better capture the gist

and accuracy of relationship prediction respectively. These enhanced object features are used to predict the final object category and pair-wise relationship in a scene graph. In the experiments, we will show that HET is indeed consistent with humans perception priority, and the generated scene graph where relationships are sorted according to their importance could be grounded to HET well.

As the structure information of HET acts as implicit guidance, we attempt to further shift the attention of the scene graph to the key relationships with explicit guidance. We employ a Relationship Ranking Module (RRM) to predict an importance score for each relationship. There are two schemes for training this module. The most direct scheme is to collect the key relationship annotation as supervision, i.e., the so-called Direct Supervision (D-Sup) scheme. As the annotation is not directly available from existing datasets, we overcome this problem by drawing support from image caption in MS-COCO (Lin et al., 2014) to extend the Visual Genome (VG) (Krishna et al., 2017) to VG-KR dataset which contains indicative annotation of key relationships. Besides, the key relationship annotation makes it possible for evaluating the performances of different models on key relationship prediction. Considering that the process of obtaining key relationship annotation may be cumbersome, we further introduce the Approximate Supervision (A-Sup) scheme based on the findings that some objective cues such as spatial scale and visual saliency, are positively correlated with relationship importance. In this way, reasonably ranking relationships is still feasible without key relationship annotation.

Preliminary version of this work has been published in Wang et al. (2020). Compared with the conference version, we have made three major extensions in this paper: First, we present a graph-based modeling method in the proposed HCP module. It models the closely correlated sibling nodes in HET as a sub-graph, and directly associates them to measure their mutual effect more precisely, achieves better message passing and feature enhancement. Second, we provide an additional approximate supervision scheme which enables the RRM training process when key relationship annotation is unavailable and thus broads its application scope. Besides, the RRM also works in a slightly different way, which uses the contextual relationship representation rather than the visual features from the backbone as input. Furthermore, we find that a Self-Attention based RRM performs better than the previous bi-directional LSTM based one. Third, more extensive experiments are conducted to verify the feasibility of the approximate supervision scheme, compare with state-of-the-art methods, and demonstrate the value of key relationship on more subsequent tasks, e.g., cross-modal retrieval.

2 Related Works

2.1 Scene Graph Generation

Scene graph generation (SGG) and Visual Relationship Detection (VRD), are two most common tasks that aim at detecting relationships between two objects. In the field of VRD, various studies (Dai et al., 2017; Li et al., 2017; Lu et al., 2016; Peyre et al., 2017; Yin et al., 2018; Yu et al., 2017; Zhang et al., 2017a, b, 2019) mainly focus on detecting each relationship independently rather than describing the structure of the scene. At the very beginning, scene graph is found useful in downstream applications including cross-modal retrieval (Johnson et al., 2015) and image captioning (Yang et al., 2019). A series of succeeding works struggle to design various approaches to improve the quality of the auto-generated scene graph. These works can be roughly categorized into two types from the perspective of whether it designs a specific model or proposes a model-agnostic method:

Specific model design Early works concentrate on designing various model structures or core message passing mechanism for gathering contextual information and improving

the features. Xu et al. (2017) employ GRU Chung et al. (2014) to pass messages between edges and nodes. Li et al. (2017) enrich the scene graph representation by introducing image caption and object information to jointly address multi-tasks. Wang et al. (2019) exploit both the object-level and relationship-level context. Zellers et al. (2018) and Tang et al. (2019) borrow the sequential or hierarchical modeling mechanism from LSTM Hochreiter and Schmidhuber (1997) or TreeLSTM Tai et al. (2015) to model the graph context. Yang et al. (2018) and Qi et al. (2019) employ the graph neural network. The widely adopted Transformer (Vaswani et al., 2017) is also found effective in this field (Koner et al., 2020). Besides, some methods borrow advantages from knowledge or commonsense (Chen et al., 2019; Gu et al., 2019; Zareian et al., 2020a, c) to assist relationship prediction. Other works like GPS-Net (Lin et al., 2020) and WSP (Zareian et al., 2020b) distinguish subject and object role of each entity, and encode two types of features for these two roles.

General framework design A recent trend of SGG is to design a model-agnostic framework or improve the loss function to tackle the long-tailed distribution problem and make the scene graph informative, since the tail fine-grained predicates are always overwhelmed by coarse ones and the scene graph is not precise enough (Tang et al., 2020). Suhail et al. (2021) creatively design the energy-based framework to suppress the head predicates based on the mutual constraint among predicates. Chiou et al. (2021) recover the unbiased distribution of predicates by dynamically estimating their frequency. Yu et al. (2021) construct a predicate tree and predict relationships in a coarse-to-fine manner. Guo et al. (2021) compute transition probability between head and tail predicates so that the head ones can be reasonably transformed into tail ones. Li et al. (2021) adopt re-sampling strategy during both image-level sampling and instance-level sampling stage. Tang et al. (2020) propose the unbiased learning framework from the perspective of causal inference.

Despite that most previous works mentioned above concentrate on fitting the annotation passively without thinking whether the relationships worth being predicted, there still exist a few works that make a meaningful attempt, filtering out some worthless relationships based on their own criteria. Liang et al. (2019) prune the dominant and easy-to-predict relationships to alleviate the annihilation problem of rare but meaningful relationships. Graph R-CNN (Yang et al., 2018) and AVR Lv et al. (2020) directly suppress relationships which are not annotated, thinking that the annotated ones are exactly the important ones. However, this is usually not the truth because of the long-tailed distribution problem in scene graph datasets (Li & Jiang, 2019; Tang et al., 2020). It is worth noting that those relationships dropped by the above works in a hard manner are not always wrong. Most of them conform to the scene content. If they are directly removed, it may damage the integrity of a scene graph. In this work, we claim that trivial relationships should not be directly eliminated, but should be arranged after the important ones instead, i.e., we eliminate the trivial relationships in a soft manner. In this way, the relationships are presented according to their importance. The most important relationships will be immediately given to convey the major content of the scene, while secondary ones are just temporarily hidden. Once more details of the scene are further required, the secondary relationships are presented.

2.2 Structured Scene Parsing

Scene graph generation can be regarded as a process of structured parsing of the scene. It is generally believed that the scene has a hierarchical structure, which means that a scene can be first decomposed into relatively independent object clusters, and the object cluster is composed of several closely related objects (Lin et al., 2016; Socher et al., 2011; Tang et al., 2022). These independent objects can continue to be decomposed into finer components. There are mainly two strategies for constructing a hierarchical structure: top-down and bottom-up strategy. Han and Zhu (2008) decompose the scene to obtain a parse graph in a top-down manner, while the scene structure in Lin et al. (2016); Socher et al. (2011) is obtained by bottom-up merging from the smallest indivisible elements. The structure is often used to assist other applications such as semantic segmentation (Sharma et al., 2015), object recognition and detection (Zhu et al., 2011), and image captioning (Yao et al., 2019), etc. In this work, we design a heuristic algorithm to build the scene structure inspired by human perception. After that, the structural information is extracted to guide SGG in an important-to-ordinary manner.

2.3 Viusal Saliency *v.s.* Gist

The so-called gist in this work refers to key relationships in a scene graph that form the major content of an image. It may cause confusion with visual saliency. Relationships about visual salient objects may be wrongly equated with the key ones. We give a detailed discussion.

An extremely rich set of studies (Hou et al., 2017; Li & Yu, 2015; Liu et al., 2018; Wang et al., 2015, 2017; Zhang et al., 2019) focus on mining visually salient objects (high contrast of luminance, hue, and saturation, center position (Itti et al., 1998; Klein & Frintrop, 2011; Xie et al., 2012), etc.). Prior works like (Pont-Tuset et al., 2020) proposes the localized caption dataset based on the idea that each word in a caption should be grounded, suggesting that there indeed exists strong association between humans attention (visually salient content) and their description (what they think important) about an image. However, it is notable that the visually salient content is not equal to that involved in the gist. He

He et al. (2019) explore gaze data and find that only 48% of fixated objects are referred in humans' descriptions about the image, while 95% of objects referred in descriptions are fixated. It suggests that contents referred in a description (i.e., contents that humans think important and should form the major events/gist) are almost visually salient and reveal where humans gaze, but what humans fixate (i.e., visually salient contents) are not always what they want to convey. Naturally, we need to emphasize that the levels in our constructed scene structure (HET) reflect the perception priority level rather than the visual saliency. Besides, this finding supports us to obtain the indicative annotations of key relationships with the help of annotated image caption.

3 Approach

A scene graph $\mathcal{G} = \{\mathcal{O}, \mathcal{R}\}$ of an image $\mathcal{I}$ contains a set of objects $\mathcal{O} = \{o_i = (c_i, \boldsymbol{b}_i)\}_{i=1}^{N}$, and their pairwise relationships $\mathcal{R} = \{r_k\}_{k=1}^{M}$, where $c_i \in \mathcal{C}$ and $\mathcal{C}$ is the set of category, $\boldsymbol{b}_i \in \mathbb{R}^4$ is the bounding box. Each r_k describes the relationship between o_i and o_j and thus contains a predicate $p_{ij} \in \mathcal{P}$, where $\mathcal{P}$ is the set of predicates.

The most widely-adopted scheme to generate the scene graph $\mathcal{G}$ is to extract the set of relationships from the image without considering the importance of each relationship. Our goal is to generate the SGoI hierarchically, whose relationships are sorted according to their importance. What's more, the sorted relationships could be grounded to the hierarchical structure in a top-down manner.

To achieve this goal, we present the Tree-Guided Importance Ranking (TGIR) model as illustrated in Fig. 3. As pointed out in Sect. 1, a scene graph should be configured with a hierarchical structure. We first devise a heuristic hierarchy construction method. The structural information of the hierarchical structure is encoded to guide the scene graph generation process. Finally, an explicit relationship importance estimation method is proposed to sort the relationships so that they can be better grounded to the hierarchical structure. Therefore, our model consists of four main modules: (1) a **Backbone** for generating object feature representations of the scene; (2) a **Hierarchical Entity Tree (HET) Construction** module for generating the hierarchical structure; (3) a **Hierarchical Contextual Propagation (HCP)** module for encoding the structural information and decoding it to generate the scene graph; (4) a **Relationship Ranking Module (RRM)** for sorting the relationships according to their importance. In the following sub-sections, we will present the detailed designs of each module.

3.1 Backbone

We adopt Faster R-CNN (Ren et al., 2015) detector as the backbone. It produces N objects and each of them has the 1024-dim visual feature $\boldsymbol{v}_i$, the category distribution vector $\boldsymbol{q}_i \in \mathbb{R}^{|\mathcal{C}|}$ and the bounding box $\boldsymbol{b}_i \in \mathbb{R}^4$. We compute the semantic representation of each object as $\boldsymbol{z}_i = \boldsymbol{W}_e^{(1)}\boldsymbol{q}_i$, where $\boldsymbol{W}_e^{(1)}$ is a word embedding matrix initialized from GloVe Pennington et al. (2014), and concatenate it with the visual feature:

$$\boldsymbol{x}_i = [\boldsymbol{v}_i; \boldsymbol{z}_i]. \tag{1}$$

The resulting feature $\boldsymbol{x}_i \in \mathbb{R}^d$ is used as the input feature in later process.

3.2 HET Construction

We aim to construct the Hierarchical Entity Tree (HET) whose top-down levels are in accordance with the perceptive levels of humans' inherent scene parsing hierarchy. Psychological works (Navon, 1977; Biederman, 2017) have shown that people perform rapid global scene analysis before conducting more detailed local object analysis, and thus a scene can be naturally represented by a perception related tree structure. It is natural that a scene can be firstly decomposed into several large objects (clusters), and then these objects (clusters) can be further decomposed into objects of smaller-scale or components (Han & Zhu, 2008). Therefore, the core idea for building HET is to arrange larger objects (clusters) on top layers of HET as far as possible. Concretely, HET is a multi-branch tree $\mathcal{T}$ with a virtual root o_0 standing for the whole image. We first sort the objects in descending order according to their spatial scale and get the sequence $\{o_{i_1}, o_{i_2}, \ldots, o_{i_N}\}$. Then the tree structure is determined by reasonably identifying the parent node of each object. For object o_{i_n}, we consider the objects with larger size $\{o_{i_m} \mid 1 \le m < n\}$, and compute the ratio

$$P_{nm} = \frac{I\left(o_{i_n}, o_{i_m}\right)}{A(o_{i_n})}, \tag{2}$$

where $A(\cdot)$ denotes the size of the object and $I(\cdot,\cdot)$ is the overlap area of two objects. In practice, we compute the overlap area of their bounding boxes. If P_{nm} is larger than a given threshold T, o_{i_m} will be a candidate parent node of o_{i_n} since o_{i_m} contains most part of o_{i_n}. If there is no candidate, the parent node of o_{i_n} is set as the root. If there are two or more

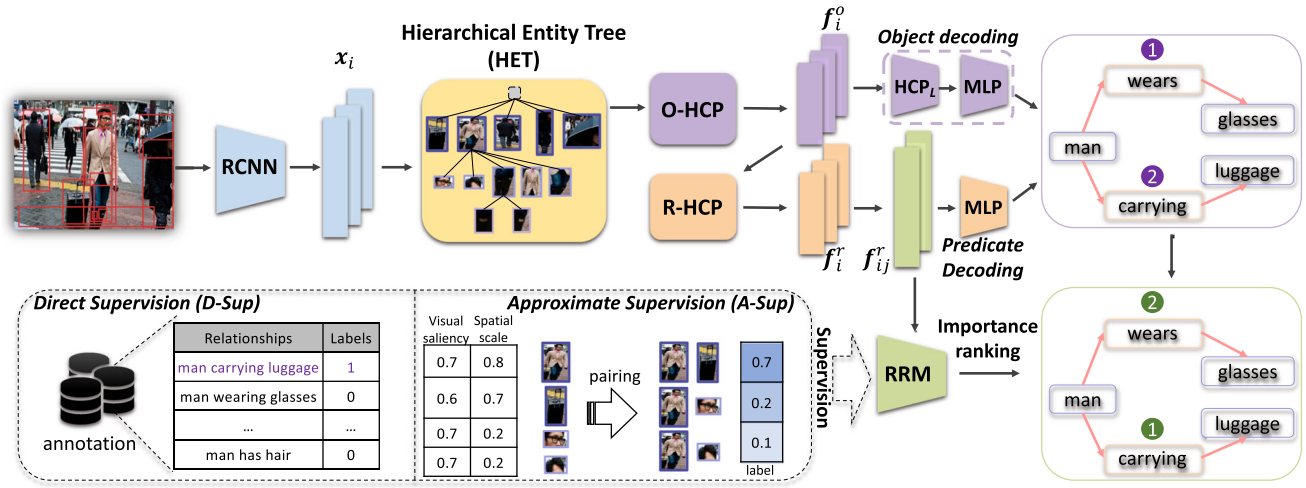


Fig. 3 The framework of our TGIR model. An object detector is applied and HET is constructed using detected objects. The object features are fed into O-HCP and R-HCP module to obtain the contextual object and relationship features which are decoded to predict the object category and predicate. Furthermore, the RRM receiving two alternative types of supervision estimates the importance to re-rank the relationships

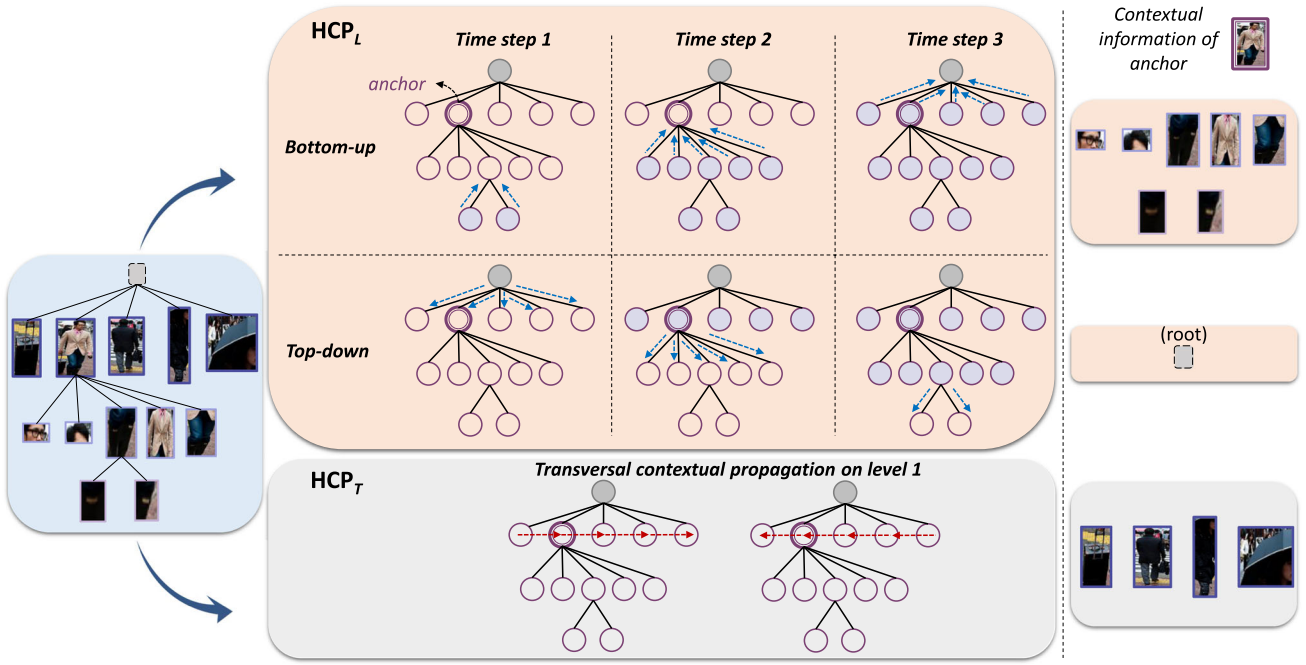


Fig. 4 An illustration of the proposed HCP module which consists of two sub-modules, HCP$_L$ and HCP$_T$. For an anchor object, HCP$_L$ broadcasts the contextual information among its ancestors and descendants in bottom-up and top-down directions, while HCP$_T$ passes messages between its sibling objects

candidates, we choose the candidate with the largest proportion of the overlap area considering that it is most likely to govern o_{i_n}. A HET example is shown in Fig. 2a. The node of *woman* governs its details such as the *bag_2* and *helmet_2*.

3.3 Hierarchical Contextual Propagation

As the levels of HET stand for the perception priority of the objects, we hope to model its structural information to guide scene graph generation. For each anchor object[1] in HET, there exists two types of objects that are most likely

[1] For convenience, we use "anchor" to refer to the target object that we consider in following parts.

to be strongly related to it. One type is the objects on the same root-to-leaf path with the anchor (i.e., its ancestors and descendants). These objects are either comprised of the anchor or subordinate to the anchor. The other type is the siblings which share a same parent with the anchor. They are on the same perceptual level with the anchor and thus humans tend to pay attention to their relationships with the anchor.

Inspired by these observations, we propose the Hierarchical Contextual Propagation (HCP) module to encode the information from two types of objects above for each anchor object, as illustrated in Fig. 4. The HCP module consists of two sub-modules, one is the Longitudinal Contextual Propagation (HCP_L) sub-module for broadcasting the contextual information along the root-to-leaf paths of HET. The other one is the Transversal Contextual Propagation (HCP_T) sub-module. It passes messages between sibling objects. We design two types of HCP_T sub-module, i.e., the bi-directional LSTM based HCP_T, and the multi-head graph attention (mGAT) based HCP_T. It consequently results in two types of HCP, denoting by HCP-B and HCP-G. We will apply HCP to encode the structural information then decode it to predict final object and predicate information which forms a scene graph. The encoding and decoding processes are based on HET and the object input features $\{\boldsymbol{x}_i\}_{i=1}^N$ obtained in Sect. 3.1. We will detail the HCP module below.

3.3.1 Longitudinal Contextual Propagation

The structure information in HET is expected to be embedded in all objects through the longitudinal contextual propagation along the vertical paths. It can be regarded as the sequential dependency modeling problem which is well-solved by LSTM Hochreiter and Schmidhuber (1997). Specifically for the tree structure, we adopt bidirectional multi-branch Tree-LSTM (Tai et al., 2015) (Bi-TLSTM). Each object has its own input feature $\boldsymbol{x}$ and an randomly initialized hidden state $\boldsymbol{h}$. For an anchor object o_t, the bidirectional process of Bi-TLSTM will collect information from its descendants (bottom direction) and ancestors (top-down direction) respectively, so that o_t receives information about its perception level in HET and those objects that may form dependent association with it.

Let $C(t)$ denote the set of children of o_t. the bottom-up direction of Bi-TLSTM will collect information from the descendants of o_t, which is formulated as follows:

$$
\begin{aligned}
\tilde{\boldsymbol{h}}_t^{\uparrow} &= \sum_{k\in C(t)} \boldsymbol{h}_k^{\uparrow},\\
\boldsymbol{i}_t^{\uparrow} &= \sigma\left(\boldsymbol{W}_{(i)}^{\uparrow}\boldsymbol{x}_t + \boldsymbol{U}_{(i)}^{\uparrow}\tilde{\boldsymbol{h}}_t^{\uparrow} + \boldsymbol{b}_{(i)}^{\uparrow}\right),\\
\boldsymbol{f}_{tk}^{\uparrow} &= \sigma\left(\boldsymbol{W}_{(f)}^{\uparrow}\boldsymbol{x}_t + \boldsymbol{U}_{(f)}^{\uparrow}\boldsymbol{h}_k^{\uparrow} + \boldsymbol{b}_{(f)}^{\uparrow}\right), \forall k \in C(t),\\
\boldsymbol{o}_t^{\uparrow} &= \sigma\left(\boldsymbol{W}_{(o)}^{\uparrow}\boldsymbol{x}_t + \boldsymbol{U}_{(o)}^{\uparrow}\tilde{\boldsymbol{h}}_t^{\uparrow} + \boldsymbol{b}_{(o)}^{\uparrow}\right),\\
\boldsymbol{u}_t^{\uparrow} &= \tanh\left(\boldsymbol{W}_{(u)}^{\uparrow}\boldsymbol{x}_t + \boldsymbol{U}_{(u)}^{\uparrow}\tilde{\boldsymbol{h}}_t^{\uparrow} + \boldsymbol{b}_{(u)}^{\uparrow}\right),\\
\boldsymbol{c}_t^{\uparrow} &= \boldsymbol{i}_t^{\uparrow}\cdot\boldsymbol{u}_t^{\uparrow} + \sum_{k\in C(t)} \boldsymbol{f}_{tk}^{\uparrow}\cdot\boldsymbol{c}_k^{\uparrow},\\
\boldsymbol{h}_t^{\uparrow} &= \boldsymbol{o}_t^{\uparrow}\cdot\tanh\left(\boldsymbol{c}_t^{\uparrow}\right),
\end{aligned}
\tag{3}
$$

where $\boldsymbol{W}_{(\cdot)}^{\uparrow}$, $\boldsymbol{U}_{(\cdot)}^{\uparrow}$, and $\boldsymbol{b}_{(\cdot)}^{\uparrow}$ are learnable parameters and σ denotes sigmoid function. $\boldsymbol{i}_t^{\uparrow}$, $\boldsymbol{f}_{tk}^{\uparrow}$ are input gate and forget gates that control the ratio of the information from children that should be received or forgotten, $\boldsymbol{o}_t^{\uparrow}$ is the output gate. The bottom-up longitudinal contextual information of o_t is saved in the updated hidden state $\boldsymbol{h}_t^{\uparrow}$. It can be used for computing longitudinal contextual information of the ancestors of o_t in following time steps, as shown in the bottom-up process in Fig. 4.

Similarly, let $p(t)$ denote the parent of o_t. The top-down direction of Bi-TLSTM will collect information from the ancestors of o_t, which is formulated as follows:

$$
\begin{aligned}
\boldsymbol{i}_t^{\downarrow} &= \sigma\left(\boldsymbol{W}_{(i)}^{\downarrow}\boldsymbol{x}_t + \boldsymbol{U}_{(i)}^{\downarrow}\boldsymbol{h}_{p(t)}^{\downarrow} + \boldsymbol{b}_{(i)}^{\downarrow}\right),\\
\boldsymbol{f}_t^{\downarrow} &= \sigma\left(\boldsymbol{W}_{(f)}^{\downarrow}\boldsymbol{x}_t + \boldsymbol{U}_{(f)}^{\downarrow}\boldsymbol{h}_{p(t)}^{\downarrow} + \boldsymbol{b}_{(f)}^{\downarrow}\right),\\
\boldsymbol{o}_t^{\downarrow} &= \sigma\left(\boldsymbol{W}_{(o)}^{\downarrow}\boldsymbol{x}_t + \boldsymbol{U}_{(o)}^{\downarrow}\boldsymbol{h}_{p(t)}^{\downarrow} + \boldsymbol{b}_{(o)}^{\downarrow}\right),\\
\boldsymbol{u}_t^{\downarrow} &= \tanh\left(\boldsymbol{W}_{(u)}^{\downarrow}\boldsymbol{x}_t + \boldsymbol{U}_{(u)}^{\downarrow}\boldsymbol{h}_{p(t)}^{\downarrow} + \boldsymbol{b}_{(u)}^{\downarrow}\right),\\
\boldsymbol{c}_t^{\downarrow} &= \boldsymbol{i}_t^{\downarrow}\cdot\boldsymbol{u}_t^{\downarrow} + \boldsymbol{f}_t^{\downarrow}\cdot\boldsymbol{c}_{p(t)}^{\downarrow},\\
\boldsymbol{h}_t^{\downarrow} &= \boldsymbol{o}_t^{\downarrow}\cdot\tanh\left(\boldsymbol{c}_t^{\downarrow}\right),
\end{aligned}
\tag{4}
$$

where $\boldsymbol{i}_t^{\downarrow}$, $\boldsymbol{f}_t^{\downarrow}$ and $\boldsymbol{o}_t^{\downarrow}$ are the input gate, forget gate, and output gate. Through these two processes, the longitudinal contextual information is saved in $\boldsymbol{h}_t^{\uparrow}$ and $\boldsymbol{h}_t^{\downarrow}$.

3.3.2 Transversal Contextual Propagation

In HET, apart from the dependent association among the objects along the root-to-leaf paths, more semantic relationships tend to exist among the sibling objects since they share similar perception priority. Therefore, this transversal context is modeled to strengthen the semantic information. The most direct way is to regard it as a sequential dependency modeling problem similar to that in Sect. 3.3.1. However, as any two sibling objects may be directly related, it is more reasonable to model them as a graph. Therefore, we propose two alternative schemes.

The first scheme is employing the bidirectional LSTM (Bi-LSTM) following (Zellers et al., 2018). For the anchor object o_t and its sibling objects, we first arrange them into a sequence. As the sequence order will not obviously influence the performance, we arrange the objects from left to right according to their horizontal coordinates of the centres of

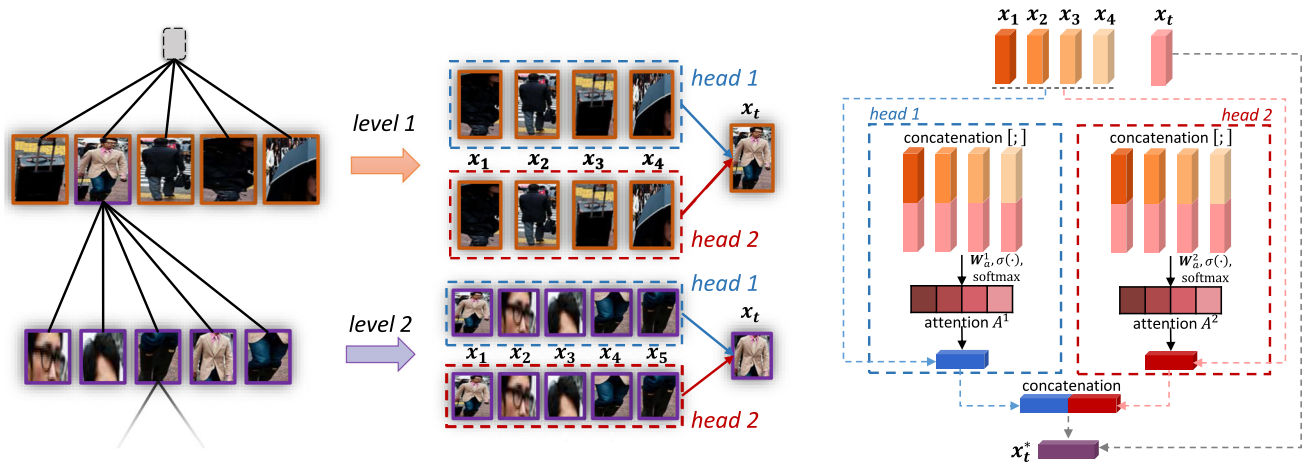


Fig. 5 An illustration of the mGAT-based HCP_T. Computation of transversal contextual feature for the anchor object o_t on level 1 and level 2 is illustrated in the middle column. Two heads are used in this figure. The right column gives a detailed computation process

the bounding boxes. Let $l(t)$ and $r(t)$ denote the left and right sibling of o_t respectively. The transversal context is constructed as:

$$\boldsymbol{h}_t^{\rightarrow} = \text{LSTM}(\boldsymbol{x}_t, \boldsymbol{h}_{l(t)}^{\rightarrow}), \quad \boldsymbol{h}_t^{\leftarrow} = \text{LSTM}(\boldsymbol{x}_t, \boldsymbol{h}_{r(t)}^{\leftarrow}), \tag{5}$$

The LSTM$(\cdot)$ denotes the standard LSTM unit (Hochreiter & Schmidhuber, 1997). The bidirectional process will collect all sibling information which is saved in $\boldsymbol{h}_t^{\rightarrow}$ and $\boldsymbol{h}_t^{\leftarrow}$.

Alternatively, as sibling objects are strongly correlated with each other, we model them as a fully connected graph and propose the multi-head graph attention (mGAT)-based (Velickovic et al., 2018) HCP_T module. It strengthens the capability of holistic relational understanding among each sub-tree in the HET.

Considering a sub-tree with the root object o_r and its children set $C(r)$, let $O_{sub} = \{o_r\} \bigcup C(r)$ denote these $N_s = |C(r)| + 1$ objects, $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{N_s}\}, \boldsymbol{x}_i \in \mathbb{R}^d$ denote their input features. We enroll the root here unless it is the virtual root of HET, because in HET, the children objects could not exist without their parent. There are multiple attention heads in mGAT-based HCP_T and each head can model specific correlation among the objects (Vaswani et al., 2017). As depicted in Fig. 5, in each head k among totally K heads, for the anchor object $o_t \in O_{sub}$ and any one object $o_j \in O_{sub}$, $j \neq t$, their correlation coefficient e_{tj}^k is computed as:

$$e_{tj}^k = \sigma\left(\boldsymbol{W}_a^k\left[\boldsymbol{W}^k\boldsymbol{x}_t; \boldsymbol{W}^k\boldsymbol{x}_j\right]\right), \tag{6}$$

where $\boldsymbol{W}^k \in \mathbb{R}^{\frac{d}{K}\times d}$, $\boldsymbol{W}_a^k \in \mathbb{R}^{1\times\frac{2d}{K}}$ are projection matrices, $[;]$ represents concatenation operation and σ denotes Leaky-ReLU function.

The correlation coefficients are normalized across objects in O_{sub} except the anchor object o_t by softmax function, then applied to these objects to obtain the aggregated feature. We concatenate the aggregated features through K heads and add it to the input feature of o_t as a residual part, which results in the final transversal contextual feature $\boldsymbol{x}_t^*$ of o_t:

$$\begin{aligned}\alpha_{tj}^k &= \text{softmax}_j(e_{tj}^k) = \frac{\exp(e_{tj}^k)}{\sum_{q\in[1,N_s],q\neq t}\exp(e_{tq}^k)},\\ \boldsymbol{x}_t^* &= \boldsymbol{x}_t + \text{Concat}_{k=1}^K\left(\sum_{j\in[1,N_s],j\neq t}\alpha_{tj}^k\boldsymbol{W}^k\boldsymbol{x}_j\right).\end{aligned} \tag{7}$$

3.3.3 Context Decoding for Scene Graph Inference

With the constructed HET and the HCP module, we describe how to decode the context encoded by HCP and inference the objects and relationships in the scene graph as illustrated in Fig. 3.

Firstly we employ a single HCP module, namely O-HCP, to model contextual information for object recognition. The input features are $\{\boldsymbol{x}_i\}_{i=1}^N$ obtained by Eq. (1) and the outputs are $\{\boldsymbol{f}_i^o\}_{i=1}^N$, where $\boldsymbol{f}_i^o$ is obtained by:

$$\begin{aligned}&\text{LSTM}-\text{based}\ \ \text{HCP}_T : \boldsymbol{f}_i^o = \boldsymbol{W}_o[\boldsymbol{h}_i^{\uparrow}; \boldsymbol{h}_i^{\downarrow}; \boldsymbol{h}_i^{\rightarrow}; \boldsymbol{h}_i^{\leftarrow}],\\ &\text{mGAT}-\text{based}\ \ \text{HCP}_T : \boldsymbol{f}_i^o = \boldsymbol{W}_o[\boldsymbol{h}_i^{\uparrow}; \boldsymbol{h}_i^{\downarrow}; \boldsymbol{x}_i^*],\end{aligned} \tag{8}$$

where $[;]$ denotes concatenation and $\boldsymbol{W}_o$ is a projection matrix. The computation process of the contextual features above follows Sects. 3.3.1 and 3.3.2.

We utilize another HCP module, namely R-HCP, to model contextual information specifically for relationship prediction. The input features are $\{\boldsymbol{f}_i^o\}_{i=1}^N$ and the outputs are denoted as $\{\boldsymbol{f}_i^r\}_{i=1}^N$.

Object Decoding With these contextual features, we decode them to predict the objects and relationships. In HET, a child object strongly depends on its parent, i.e., information of the parent object is helpful for prediction of the child object. Therefore, we employ a single direction HCP_L submodule as a decoder to predict objects in a top-down manner, as illustrated in Fig. 6. For the object o_i, the decoder receives information from its parent, using $[\boldsymbol{f}_i^o; \boldsymbol{W}_e^{(2)}\boldsymbol{q}_p]$ as input, where $\boldsymbol{W}_e^{(2)}$ is a word embedding matrix and $\boldsymbol{q}_p$ is the predicted category distribution vector of the parent of o_i. The output feature representation $\boldsymbol{h}_i^{dec}$ is fed into an MLP classifier to predict the category of o_i together with the confidence s_i.

Predicate Decoding When predicting the predicate p_{ij} between o_i and o_j, we first gather the features of o_i and o_j as:

$$\boldsymbol{f}_{ij}^g = [\boldsymbol{W}_r \boldsymbol{f}_i^r; \boldsymbol{W}_r \boldsymbol{f}_j^r], \tag{9}$$

where $\boldsymbol{W}_r$ is a projection matrix. On the other hand, we use the union box of o_i and o_j to extract the visual feature $\boldsymbol{f}_{ij}^v$ from the image feature map with RoI pooling operation. We also follow DRNet (Dai et al., 2017) to build the spatial feature $\boldsymbol{f}_{ij}^s$. The final relationship feature is obtained as:

$$\boldsymbol{f}_{ij}^r = \boldsymbol{f}_{ij}^g \circ (\boldsymbol{f}_{ij}^v + \boldsymbol{f}_{ij}^s), \tag{10}$$

where $\circ$ denotes element-wise multiplication. The relationship feature is fed into another MLP classifier to predict predicate and its corresponding confidence s_{ij}. The confidence α_{ij} of the relationship between o_i and o_j is computed as the multiplication of the confidence of subject, object, predicate:

$$\alpha_{ij} = s_i \cdot s_j \cdot s_{ij}. \tag{11}$$

The confidence is used to sort the relationship.

3.4 Relationship Ranking Module

So far, we have obtained objects and relationships in the scene graph under the guidance of HET. During inference, to achieve our goal that a scene graph is given in an importance-first manner, the detected relationships should be manually attached to HET and present them exactly following the root-to-leaf paths of HET, i.e., firstly giving the relationships among objects from the top level, then giving those among objects from the second level, etc. In this way, we can obtain an importance-first relationship sequence. However, this process is not fully automatic. The model introduced before this sub-section could only provide the relationship sequence by sorting the relationships according to the confidence α_{ij}. This sequence may be not importance-first because only implicit structural information is considered when computing α_{ij}. If we can evaluate the importance of each relationship explicitly and enroll it into α_{ij}, the model will be able to automatically generate the importance-first relationship sequence which is consistent with that derived from HET.

In this sub-section, we further employ a Relation Ranking Module (RRM) to explicitly estimate the relationship importance. We intend to encode the global context among all the relationships so that the importance of a certain relationship can be reasonably adjusted considering the importance of other relationships. Therefore, we feed the relationship feature $\boldsymbol{f}_{ij}^r$ into an importance head denoted by Φ, which is implemented as a Self-Attention (SA) module (Vaswani et al., 2017). The importance of the relationship is obtained as:

$$z_{ij} = \text{LN}(\Phi(\boldsymbol{f}_{ij}^r)), \tag{12}$$

where LN denotes a fully connected layer. During inference, we enroll z_{ij} into α_{ij} so that the resulting α_{ij}^* for sorting relationships takes importance into consideration:

$$\alpha_{ij}^* = \alpha_{ij} \cdot z_{ij}. \tag{13}$$

RRM Training There still exists a vital problem: how to obtain effective supervision to train RRM, as there is no directly available supervision. We propose two alternative schemes. The first one is the so-called Direct Supervision (D-Sup) scheme, i.e., collecting the key relationship annotation as supervision. The key relationships are collected with the assistance of image caption, based on the thinking that an image caption almost reveals the most important content in the image as described in Sect. 2. We will introduce the annotation collection process and the resulting VG-KR dataset in Sect. 4.1.

Nevertheless, as the collection process above may be cumbersome, we further propose the Approximate Supervision (A-Sup) scheme, i.e., we intend to replace the key relationship annotation with some readily available but effective indicators of relationship importance. Intuitively, *visual saliency* and *spatial scale* maybe suitable cues. As analyzed in Sect. 2, regions of humans' interest can be tracked under the guidance of visual saliency, although they do not always form the major events that humans want to convey. Two examples are shown in Fig. 7a, b. Besides, the spatial scale is also an important reference for estimating the perceptive level of objects, as shown in Fig. 7c. We conduct preliminary statistical analyses to verify whether these two cues are qualified indicators.

In the field of psychology, perceptual salience is proposed, which is the cognitive bias that predisposes individuals to focus on items that are more prominent or emotionally striking and ignore those that are unremarkable, even though this

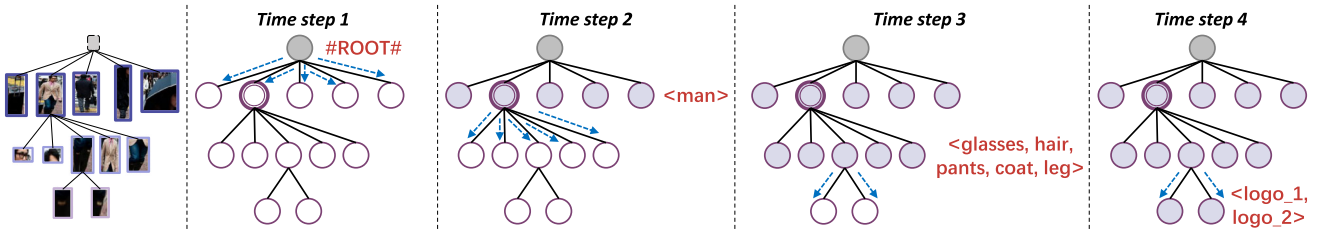


Fig. 6 An illustration of the top-down decoding process for predicting objects. In each time step, an object node receives information from its parent and its category is predicted. Only the categories of objects on the same root-to-leaf paths with the second object on level 1 (i.e., ⟨man⟩) are shown in this figure

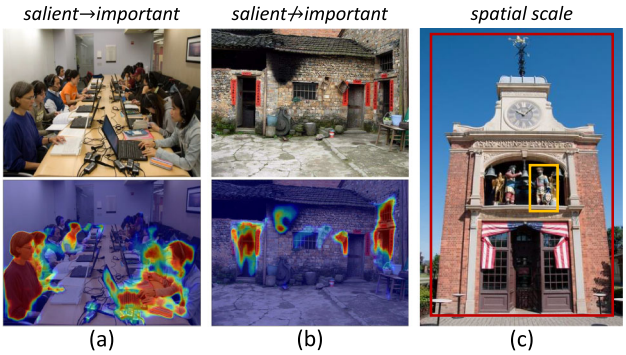


Fig. 7 Visually salient objects maybe important in some circumstances as shown in (**a**), while may just catch one's eyesight but will not be thought important in other cases such as (**b**). The saliency maps in the bottom of (**a**) and (**b**) are computed by Deng et al. (2018). Object with large spatial scale (red box) is usually more important than the smaller object (yellow box), as shown in (**c**)

difference is often irrelevant by objective standards. Kahneman et al. (1982); Bordalo et al. (2012) inspired by these works, we put forward the so-called *cognitive saliency* (CS) to measure the importance of the relationship from humans' perspective. Considering the measurement of CS of a specific relationship, we employ *its times being referred within the five captions* of each image, which can be directly obtained from our collected VG-KR dataset (originally from MS-COCO). Here, the total number of captions of each image in VG-KR is 5. Naturally the value of CS ranges discretely from 1 to 5. The spatial scale and visual saliency of a relationship $r_k = \langle o_i, p_{ij}, o_j \rangle$ are defined as $u_i + u_j$ and $v_i + v_j$ respectively, where u_i and u_j denote the normalized area of the bounding box of o_i and o_j, v_i and v_j denote the visual saliency of o_i and o_j (the average saliency of the pixels inside the bounding box which is computed by advanced salient object detection methods, such as R3Net (Deng et al., 2018)). If a cue is a qualified indicator, it should be proportional to CS. We randomly sample 50,000 key relationships from VG-KR and the process is repeated 3 times. In Fig. 8a and b, we draw the line charts whose Y-axis is CS and X-axis is spatial scale or visual saliency. They are obtained by dividing the continuous value space of spatial scale or visual saliency into 50 intervals and averaging the CS values in each interval. From these two line charts, we observe that spatial scale is a qualified indicator, while visual saliency may not strictly meet the standard because as the visual saliency of a relationship goes up, the CS drops, i.e., the relationship with large visual saliency is not as important as expected. To better understand the reason, we further extract the relationships with the first quarter of spatial scale or visual saliency and analyze the proportion of each relationship, as shown in Fig. 8c and d. It is observed that a large part of relationships with large visual saliency are those between an independent object and its components, such as *hand of man*, *letter on sign*, and *man wearing tie*. Actually, these relationships are indeed not so image-specific and convey little information. Humans will generally overlook them. However, if the visual saliency of an object is large, the visual saliency of its components will be also large. It explains the phenomenon when the visual saliency keeps increasing, the CS decreases.

From preliminary analyses above, we know that visual saliency and spatial scale are positively correlated with the relationship importance to a certain extent, but the visual saliency should not be used independently. Consequently, in our A-sup scheme, we propose to approximate the supervision using these two simple but effective cues. For each object o_i, we compute its normalized spatial u_i and visual saliency v_i as described above. For relationship r_k corresponding to o_i and o_j, its importance is approximated as:

$$\psi_{ij} = (1 - \lambda)(u_i + u_j) + \lambda(v_i + v_j), \tag{14}$$

where λ is a combination factor and its value will be discussed in Sect. 4.2.3. The ψ_{ij} of all relationships in the image are collected and normalized to form a vectorized probabilistic label $\boldsymbol{\Psi}$.

3.5 Optimization

The loss function for model optimization is divided into two parts. The first part is the loss for object and predicate classification. We adopt the cross-entropy loss. Let y_i^o and y_{ij}^r denote the ground truth category of object and predicate, and

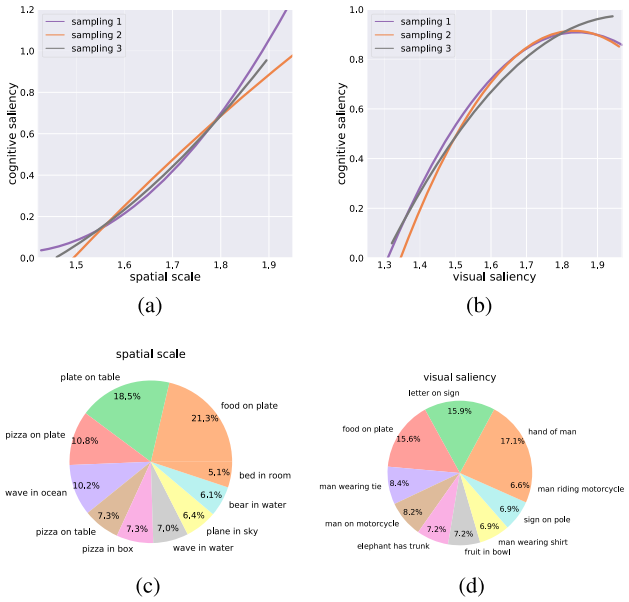


Fig. 8 Analyses of spatial scale and visual saliency. (**a**)∼(**b**) are the line charts which demonstrate the association between the cognitive saliency (CS) and spatial scale / visual saliency. (**c**)∼(**d**) show the proportion of each relationship with the first quarter of spatial scale/visual saliency

$\boldsymbol{p}_i^o$ and $\boldsymbol{p}_{ij}^r$ denote the predicted distribution of object and predicate respectively. The $\boldsymbol{p}_i^o(y_i^o)$ and $\boldsymbol{p}_{ij}^r(y_{ij}^r)$ are the predicted probability of the ground truth categories. The loss is defined as:

$$\mathcal{L}_{cls} = -\frac{1}{Z_1}\sum_i \log \boldsymbol{p}_i^o(y_i^o) - \frac{1}{Z_2}\sum_{i\neq j} \log \boldsymbol{p}_{ij}^r(y_{ij}^r). \tag{15}$$

The second part is the loss for training RRM. When using D-Sup scheme, as the annotation is binary, i.e., a relationship is either key or not, we use the binary cross entropy loss. Let $y_{ij} \in \{0, 1\}$ denote the ground truth. The loss function is:

$$\mathcal{L}_{key}^D = -\frac{1}{Z_3}\sum_{i\neq j}(y_{ij}\log\sigma(z_{ij}) + (1-y_{ij})\log(1-\sigma(z_{ij}))), \tag{16}$$

where σ denotes the sigmoid function.

As for A-Sup scheme, we minimize the KL-divergence between the approximated importance, i.e., the probabilistic label $\boldsymbol{\Psi} \in \mathbb{R}^M$ (M is the number of relationships in an image) and the predicted importance $\boldsymbol{z} \in \mathbb{R}^M$ (a vector consisting of z_{ij} of all relationships in an image):

$$\mathcal{L}_{key}^A = KL\left(\text{softmax}(\boldsymbol{z}) \,||\, \boldsymbol{\Psi}\right). \tag{17}$$

The overall loss function is:

$$\mathcal{L} = \mathcal{L}_{cls} + \gamma\mathcal{L}_{key}. \tag{18}$$

In the equations above, Z_1, Z_2, and Z_3 are normalization factors, whose values are the numbers of items that contribute to the loss function in a batch. γ denotes balance factor, which is set to 1 and 1000 for D-Sup and A-Sup scheme empirically.

4 Experiments on Scene Graph Generation

In this section, we extensively evaluate our TGIR model on three datasets. We first analyse the effect of individual modules, and then we compare our method with the state-of-the-art methods.

4.1 Experimental Settings

Datasets: We verify the effectiveness of the proposed method and compare with other state-of-the-art methods on two widely used public datasets as well as our collected datasets:

(1) *VRD* (Lu et al., 2016) is the benchmarking dataset for visual relationship detection task, which contains 4000/1000 training/test images and covers 100 object categories and 70 predicate categories.

(2) *Visual Genome (VG)* is a large-scale dataset with rich annotation of objects, attributes, and pairwise relationships, containing 75,651/32,422 training/test images. During the training stage, 5,000 images are split from the training set for validation. We adopt the most widely used version of VG, namely VG150 (Xu et al., 2017), which covers 150 object categories and 50 predicates.

(3) *VG200* and *VG-KR* are our collected datasets containing the indicative annotation of key relationships based on VG. We associate the relationships referred in the annotated image captions in MS-COCO with those from VG. Concretely, there are 51,498 images which belong to both VG and MS-COCO. Therefore, the annotation about the relationship and image caption on these images is available. They form the subset named *VG-COCO* (*VGC*) and we conduct three-stage processing on it. (i) Stanford Scene Graph Parser (Schuster et al., 2015) is used to extract the relationships from the image captions. They make up the set of key relationships, denoted by $\mathcal{R}^{\mathcal{K}}$. (ii) We next cleanse the raw relationship annotation of *VGC* following Xu et al. (2017), keep the most frequent 150 object categories and 50 predicates, and add another most frequent 50 object categories and 30 predicates in $\mathcal{R}^{\mathcal{K}}$, in order to keep as many key relationships as possible for the following third step. After dropping images without relationships in *VGC*, we get a new subset VG200 (i.e. 200 object categories) which contains 46,562

Table 1 Statistics of VG200, VG-KR, and VG150. #Img. denotes "number of images". Rel. denotes "relationship"

Dataset	Images	#Img. with rel.	Object categories	Predicate categories	Object instances	Rel. instances	Key rel. instances	#Img. with key rel. instances
VG200	51,498	46,562	200	80	619,119	442,425	101,312	26,992
VG-KR	26,992	26,992	200	80	360,306	250,755		
VG150	108,073	89,169	150	50	1,145,398	622,705	–	–

Captions:
1. A man walking along the beach while holding a surfboard.
2. A man on a beach holding a surfboard.
3. A man holding a colorful surfboard going towards the ocean.

Relationships:
<man, holding, surfboard>, <man, on, beach>, <man, wearing, pant>, <leaf, on, surfboard>, <sand, on, surfboard>, <water, behind, wave>, <head, of, man>

Captions:
1. A person riding a bike with a dog in a basket.
2. A person and a dog on a bike.
3. A man riding a bicycle with a dog in a basket on the back.

Relationships:
<dog, in, basket>, <dog, on, bike>, <man, riding, bike>, <basket, on, bike>, <car, near, building>, <tree, near, car>, <man, holding, bag>

Captions:
1. A person riding a bike holding a large plate of bread on his head.
2. A person that is riding a bike in the street.
3. A person riding a bike with a tray of sandwiches on his head.

Relationships:
<man, riding, bike>, <man, carrying, bread>, <man, holding, tray>, <tray, covered in, bread>

Fig. 9 Some examples from VG-KR dataset. Each image is shown with 3 captions and ground truth relationships. The purple ones are key relationships which correspond to the red contents in captions

images. (3) Finally, we associate $\mathcal{R}^{\mathcal{K}}$ with the relationships in VG200. The relationships in VG200 whose subject and object WordNet synsets (Miller, 1992) can be aligned with those of relationships in $\mathcal{R}^{\mathcal{K}}$ are marked as key ones. After filtering out the images without key relationships in VG200, here comes the VG-KR which contains 26,992 images. For both VG200 and VG-KR, we split the training and test set by 7:3 ratio, leading to 32,510/14,052 training/test images in VG200, and 18,720/8,272 training/test images in VG-KR. We additionally split 5,000 images from the training set for validation.

We show several examples in Fig. 9 and give the detailed statistics of our datasets in Table 1. The VG200 and VG-KR contain richer categories, predicates, as well as object and relationship instances per image compared to VG150. Moreover, VG-KR contains indicative annotation of key relationships. In Fig. 10a, we show the distribution of images that contain different numbers of key relationships. More than 90% of images contain less than 7 key relationships. Given each predicate, we explore its probability to make up a key relationship, as shown by the red bars of Fig. 10c. The predicates with large probability to be key ones, such as *throwing*, *brushing*, and *sniffing* are usually verbs with rich semantics, and they are the tailed predicates under a long-tailed predicate distribution (Tang et al., 2020) as illustrated by the blue line in Fig. 10c. They are image-specific and when we see these predicates, a scene can be roughly imagined. While predicates like *near*, *has*, and *of*, which are the head predicates, are less likely to make up the key relationships. To better understand the difference of key and trivial relationships, we give three examples in Fig. 10b. Usually, the relationship *man throwing frisbee* is the main content of an image, but the relationship *head belonging to guy* would be seldom mentioned, and *branch growing on tree* is usually secondary.

Evaluation: Conventional evaluation for scene graph generation follows triplet-match rule, i.e., only if three components of a triplet match the ground truth will it be a correct one. We adopt three universal protocols (Xu et al., 2017): PREDCLS, SGCLS, and SGGEN. Both Recall (R@K) (Xu et al., 2017) and mRecall (mR@K) (Chen et al., 2019; Tang et al., 2019) metrics are used where K is set to 20, 50, and 100. Previous works mainly report R@K while recent studies prefer mR@K which better balances the performance of each predicate. When evaluating key relationship prediction (on VG-KR), we only adopt the PREDCLS protocol to eliminate the interference of errors from object detector, and add a tuple-match rule (only subject and object are required to match the ground truth) to investigate the ability to find

Fig. 10 Statistical analysis on VG-KR. **a** The distribution of images that contain different numbers of key relationships. **b** The distribution of the key relationships consisting of a given predicate. The predicates *throwing*, *belonging to*, and *growing on* are shown. **c** Each red bar stands for the probability that a predicate constitutes a key relationship, which is the ratio of the number of key relationships and that of all relationships consisting of this predicate. The blue line represents the frequency of the predicates

proper pairs. Meanwhile, as the number of key relationships are much less, K is set to a relatively small value, i.e., 5, 10, and 20. When it comes to the VRD dataset which is for visual relationship detection task, we follow its literature and additionally adopt PHRDET protocol (Yu et al., 2017) which allows to predict multiple predicates (i.e., k=1, 10, or 70) for each pair.

Implementation Details: All models are implemented with the open source platform PyTorch.[2] In our model, the dimension of hidden features in HCP_L and HCP_T is 512. The mGAT-based HCP_T uses $K = 8$ heads. The contextual features $\{\boldsymbol{f}_i^o\}_{i=1}^N$ and $\{\boldsymbol{f}_i^r\}_{i=1}^N$ are of 1024 dimension. The final relationship feature $\boldsymbol{f}_{ij}^r$ is of 4096 dimension. The GloVe embeddings $\boldsymbol{W}_e^{(1)}$ and $\boldsymbol{W}_e^{(2)}$ are of 200 dimension. To eliminate wrong hierarchical association in HET, we set the threshold T to 0.9. Following the literature, we train a Faster R-CNN object detector (Ren et al., 2015) with ResNeXt101 (Xie et al., 2017) backbone for all models, which is pretrained on MS-COCO, then finetuned on VG150, VG200, and VRD for evaluation on VG150, VG-KR, and VRD respectively. The detector is frozen after finishing training. To train the scene graph generation model, we exploit SGD optimizer with the initial learning rate set to 0.02 and the batch size set to 32. The training process lasts for 12 epochs and the learning rate decreases by a factor of 10 at the 7th and 10th epoch respectively.

[2] Source code and our collected dataset are available at https://github.com/Kenneth-Wong/TGIR.

4.2 Ablation Study

In this section, we will firstly evaluate the rationality of HET, then evaluate the effectiveness of HCP, and explore the best configuration of RRM and A-Sup scheme.

4.2.1 Rationality Evaluation of HET

As HET is constructed heuristically and there is no ground truth, we begin with validating whether HET has a potential to reveal humans' perceptive habits, i.e., the relationships between the top objects on HET are indeed the key ones, in an indirect way. We train two models, HCP+RRM and HCP, and compare the depth distribution of top-5 predicted relationships of these two models. The depth of a relationship is represented by tuple (d_{o_i}, d_{o_j}) consisting of the depth of subject and object, and the depth of root is defined as 1. As shown in Fig. 11a, we observe a significant increment on the ratio of the depth (2, 2) and (2, 3), and a drop on (3, 3) when adding RRM. It implies that relationships which are closer to the root of HET are favoured by RRM. We also analyze the association between the confidence α_{ij}^* (Eq. 13) and the depth of the relationship. We sample 10,000 relationships from each depth predicted by HCP+RRM model five times. In Fig. 11b, the confidence decreases as the depth increases. Therefore, different levels of HET indeed indicate different perceptive importance of relationships. This characteristic makes it convenient to reasonably adjust the scale of a scene graph. When the scale of a scene graph is expected to be restricted, it is feasible for our scene graph by cutting off some deep branches of HET which are usually trivial relationships, but it is difficult to realize in an ordinary scene graph.

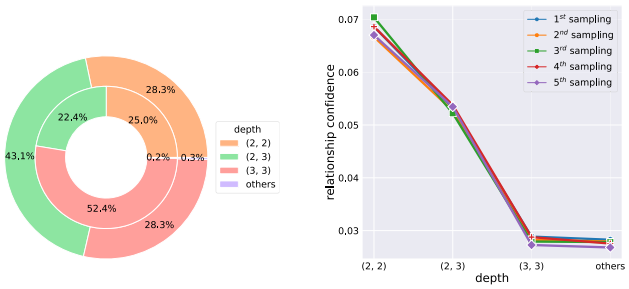


Fig. 11 Analyses about the depth. **a** Depth distribution of top-5 predicted relationships. The results of the inner and outer rings are from HCP and HCP+RRM respectively. **b** The association between the confidence and depth of the relationship

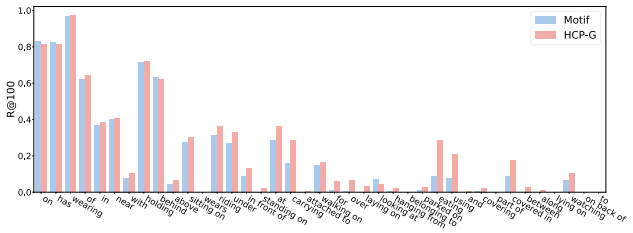


Fig. 12 Per-predicate recall on different predicates. Predicates are sorted in descending order from left to right according to the frequency

4.2.2 Effectiveness Evaluation of HCP

We conduct ablative experiments to validate the effectiveness of the sub-modules of HCP, i.e., HCP_L and HCP_T. Besides, we compare Bi-LSTM and mGAT mechanism. When the HCP configures with a Bi-LSTM-based HCP_T sub-module, we denote it by HCP-B, otherwise we denote it as HCP-G configured with an mGAT-based HCP_T sub-module. We do not apply RRM. Experiments are conducted on VG150 and VRD and the results are shown in Table 2. We additionally provide the result of Motif (Zellers et al., 2018) since when ablating HCP_L, the remaining HCP_T would almost degenerate to Motif.

Comparing HCP_L with HCP-B/HCP-G, it can be inferred that HCP_T contributes to both R@K and mR@K since it directly propagates the contextual information between all possibly related objects, and restricts this information propagation process among the objects which do not share the same parent so that those weak relationships are removed.

Comparing Motif with HCP-B/HCP-G, HCP_L which encourages the longitudinal contextual propagation especially benefits mR@K. We further compare the predicate-wise recall of Motif and HCP-G in Fig. 12 where the predicates are sorted in an descending order from left to right according to the frequency. It is noted that the improvements of mR@K mainly owe to the improved performance on tailed predicates. As indicated in Fig. 10c and Sect. 4.2.1, the relationships consisting of these tailed predicates are usually the key relationships and they exist between the top objects in HET, which means that the HCP_L benefits the key relationships prediction even without extra supervision.

4.2.3 Evaluation of RRM and A-Sup

We conduct ablative experiments to explore the best configuration of supervision approximation in the A-Sup scheme. We adjust λ in Eq. (14), ranging from 0 to 1 uniformly, to balance the weights of spatial scale and visual saliency. Meanwhile, we verify the necessity of global context modeling considered in the design of RRM. When ranking the relationships, we employ the Self-Attention (SA) module to model global context of relationships so that the importance of each relationship could be estimated with consideration of other relationships. Oppositely, we replace it with a simple two-layer MLP or an LSTM which is weaker than SA on context modeling. This ablation study is conducted on Motif. We report the improvement on R@K and mR@K on VG-KR dataset after adding RRM, and the improvement is averaged over different K (K is set to 1, 5, 10, and 20). The results are shown in Fig. 13.

We make two observations: First, as λ increases, the weight of visual saliency also increases, but the performances tend to drop. Especially when λ is larger than 0.7, the negative impact comes into prominence (the improvement is near or below 0.0%). This phenomenon is obvious on R@K. Considering the performance of SA on mR@K, the positive impact first reaches a peak at about $\lambda = 0.1$ then drops as λ keeps increasing. It suggests that visual saliency is not an ideal relationship importance indicator compared with spatial scale, which is consistent with findings from previous statistical analyses (Sect. 3.4). Second, SA basically outperforms MLP and LSTM with a relatively small λ (as λ becomes larger, the quality of the approximate supervision declines which makes this superiority becomes not so obvious), indicating the necessity of modeling global context when ranking relationships. Therefore, to balance R@K and mR@K metrics, we set $\lambda = 0.1$ in following experiments so that the SA-based RRM improves the performances as significant as possible on both of these two metrics.

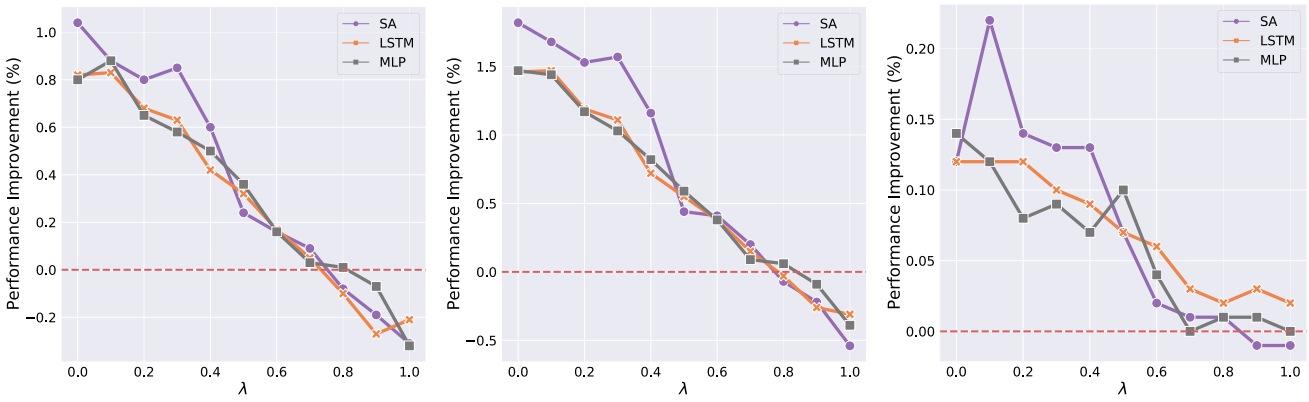
4.3 Comparison with State-of-the-Arts

In this section, we compare our proposed method with other state-of-the-art methods on conventional scene graph generation/visual relationship detection tasks. Additionally, as key relationships are especially valuable, we further compare these methods on key relationship prediction task.

Table 2 Performances (%) of different ablative models under the PREDCLS protocol

Dataset	Model	R@20/50/100	mR@20/50/100
VG150	Motif Zellers et al. (2018)	60.0/66.2/67.5	11.4/14.0/14.9
	HCP_L	58.4/64.6/66.7	10.9/13.4/14.2
	HCP-B	60.1/66.3/67.6	11.7/14.3/15.1
	HCP-G	**60.2/66.4/67.7**	**12.7/16.1/17.2**
VRD	Motif Zellers et al. (2018)	54.0/56.6/56.7	8.2/9.1/9.2
	HCP_L	54.3/56.6/56.7	8.9/9.6/9.7
	HCP-B	54.6/56.8/56.9	9.0/9.8/9.9
	HCP-G	**54.6/57.0/57.0**	**9.3/10.2/10.3**

The best performances are shown in bold



(a) R@K (triplet-match rule) (b) R@K (tuple-match rule) (c) mR@K (triplet-match rule)

Fig. 13 The performance improvement after adding RRM under various configurations of A-Sup scheme. Results are obtained under PREDCLS protocol

4.3.1 Comparison on Conventional Tasks

For comparison on VG150, we compare our TGIR model with following state-of-the-art methods: KERN Chen et al. (2019), IMP Xu et al. (2017), MemNet Wang et al. (2019), Motif Zellers et al. (2018), VCTree Tang et al. (2019) and Seq2Seq Lu et al. (2021). These methods design a specific model as introduced in Sect. 2 and we call them **TYPE-I** methods. We denote our model configured with Bi-LSTM-based HCP_T as **TGIR-B** and that configured with mGAT-based HCP_T as **TGIR-G**. All of them can be divided into several key components, including the non-core modules (e.g., the object detector, relationship instance sampling module, and the visual feature extractor), and the most important core scene graph inference module (e.g., the HCP is our core scene graph inference module) that most previous works focus on for improving. However, some works (Tang et al., 2020) have found that the improvement of the non-core modules can also achieve better scene graph generation performances, and this positive effect is even significant than that from the core scene graph inference module. Actually, these non-core modules can be easily migrated among different methods. In order to compare different methods as fair as possible and especially compare different core scene graph inference modules, referring to Motif (Zellers et al., 2018), we re-implement some of these methods using the same implementation scheme for the non-core modules.[3] All models are only trained with the basic classification loss $\mathcal{L}_{cls}$ as described in Eq. (15). On the other hand, we also compare with recent works which propose a general framework, including TDE Tang et al. (2020), PCPL Yan et al. (2020), CogTree Yu et al. (2021), DLFE Chiou et al. (2021), and BASGG Guo et al. (2021). We call them **TYPE-II** methods.

In Table 3, we list the performances of TYPE-I methods and TYPE-II methods, and report the results of some TYPE-I methods re-implemented by ourselves. It can be seen that our re-implemented versions obtain obvious improvement on R@K, and even the early work IMP is competitive among these methods. Another observation is that when adopting the same non-core modules, the performance gap among these methods has narrowed, which reflects the real gap among their core scene graph inference modules. Besides, TYPE-II methods focus on improving the recall of tailed predicates and thus achieve better mR@K performances, but it usually

[3] There exist some differences between the performances of our conference version and this paper.

Table 3 Performances (%) on VG150

		SGGEN		SGCLS		PREDCLS	
		R@20/50/100	mR@20/50/100	R@20/50/100	mR@20/50/100	R@20/50/100	mR@20/50/100
TYPE-I	MemNet Wang et al. (2019)	7.7/11.4/13.9	–/–/–	23.3/27.8/29.5	–/–/–	42.1/53.2/57.9	–/–/–
	IMP Xu et al. (2017)	–/3.4/4.2	–/–/–	–/21.7/24.4	–/–/–	–/44.8/53.0	–/–/–
	IMP†Xu et al. (2017); Zellers et al. (2018); Chen et al. (2019)	14.6/20.7/24.5	–/3.8/4.8	31.7/34.6/35.4	–/5.8/6.0	52.7/59.3/61.3	–/9.8/10.5
	KERN Chen et al. (2019)	–/27.1/29.8	–/6.4/7.3	–/36.7/37.4	–/9.4/10.0	–/65.8/67.6	–/17.7/19.2
	FREQ Zellers et al. (2018); Tang et al. (2019)	20.1/26.2/30.1	4.5/6.1/7.1	29.3/32.3/32.9	5.1/7.2/8.5	53.6/60.6/62.2	8.3/13.0/16.0
	Motif Zellers et al. (2018); Tang et al. (2019)	21.4/27.2/30.3	4.2/5.7/6.6	32.9/35.8/36.5	6.3/7.7/8.2	58.5/65.2/67.1	10.8/14.0/15.3
	VCTree Tang et al. (2019)	22.0/27.9/31.3	5.2/6.9/8.0	35.2/38.1/38.8	8.2/10.1/10.8	60.1/66.4/68.1	14.0/17.9/19.4
	VCTree†Tang et al. (2019, 2020)	24.8/31.8/36.1	4.9/6.6/7.7	35.4/38.9/39.8	6.3/7.5/8.0	59.1/65.5/67.4	12.4/15.4/16.6
	Seq2Seq Lu et al. (2021)	22.1/30.9/34.4	7.5/9.6/12.1	34.5/38.3/39.0	11.9/14.7/16.2	60.3/66.4/68.5	21.3/26.1/30.5
TYPE-II	TDE Tang et al. (2020)	12.4/16.9/20.3	5.8/8.2/9.8	21.7/27.7/29.9	9.8/13.1/14.9	33.6/46.2/51.4	18.5/25.5/29.1
	PCPL Zellers et al. (2018); Yan et al. (2020); Chiou et al. (2021)	21.3/27.8/31.7	8.0/10.7/12.6	31.9/35.3/36.1	9.9/12.0/12.7	48.4/54.7/56.5	19.3/24.3/26.1
	CogTree Zellers et al. (2018); Yu et al. (2021)	15.7/20.0/22.1	7.9/10.4/11.8	19.4/21.6/22.2	12.1/14.9/16.1	31.1/35.6/36.8	20.9/26.4/29.0
	DLFE Zellers et al. (2018); Chiou et al. (2021)	18.9/25.4/29.4	8.6/11.7/13.8	29.0/32.3/33.1	12.8/15.2/15.9	46.4/52.5/54.2	22.1/26.9/28.8
	BASGG Guo et al. (2021)	16.8/23.0/26.9	10.7/13.5/15.6	26.9/30.1/31.0	14.0/16.5/17.5	44.4/50.7/52.5	24.8/29.7/31.7
	IMP‡	19.9/26.2/29.9	3.1/4.2/4.8	34.5/37.7/38.4	5.7/7.0/7.4	58.9/65.7/67.1	9.9/12.5/13.4
	KERN‡	22.3/30.1/35.1	3.7/5.2/6.2	34.0/37.2/37.9	5.8/6.9/7.2	59.1/65.8/67.2	10.2/13.0/13.8
	Motif‡	22.0/29.8/34.7	3.7/5.2/6.2	34.9/38.0/38.5	6.9/8.2/8.6	60.0/66.2/67.5	11.4/14.0/14.9
	VCTree‡	**23.2/30.6/35.1**	3.4/4.7/5.6	**35.3/38.3/38.9**	*7.4/9.0/9.4*	**60.5/66.6/67.9**	*12.2/15.0/15.9*
	TGIR-B	*23.1/30.3/35.0*	*4.2/5.5/6.4*	*35.2/38.2/38.8*	7.1/8.4/8.7	60.1/66.3/67.6	11.7/14.3/15.1
	TGIR-G	21.3/29.1/34.8	**4.4/6.4/7.9**	**35.3/38.3/38.9**	**7.6/9.2/9.7**	*60.2/66.4/67.7*	**12.7/16.1/17.2**

The top-2 performances are shown with bold and italicized respectively
The † means the results are from the re-implemented version of the third-party works, and the ‡ stands for our re-implemented results

Table 4 Performances (%) on VRD

	PREDCLS	SGCLS	RELDET/SGGEN			PHRDET		
	k=1	k=1	k=1	k=10	k=70	k=1	k=10	k=70
VRD Lu et al. (2016)	–/47.9/47.9	–	–/16.2/17.0	–	–	–/13.9/14.7	–	–
PPRFCN Zhang et al. (2017b)	–/47.4/47.4	–	–/19.6/23.2	–	–	–/14.4/15.7	–	–
VTranse Zhang et al. (2017a)	–/44.8/44.8	–	–/19.4/22.4	–	–	–/14.1/15.2	–	–
ViP Li et al. (2017)	–	–	–/17.3/20.0	–	–	–/22.8/27.9	–	–
VRL Liang et al. (2017)	–	–	–/18.2/20.8	–	–	–/21.4/22.6	–	–
Zoom-Net Yin et al. (2018)	–/50.7/50.7	–	–/18.9/21.4	–	–/21.4/27.3	–/24.8/28.1	–	–/29.1/37.3
CAI+SCA-M Yin et al. (2018)	–/56.0/56.0	–	–/19.5/22.4	–	–/22.3/28.5	–/25.2/28.9	–	–/29.6/38.4
KL-Dist Yu et al. (2017)	–/55.2/55.2	–	–/19.2/21.3	–/22.6/29.9	–/22.7/31.9	–/23.1/24.0	–/26.5/29.8	–/26.3/29.4
RELDN Zhang et al. (2019)	–	–	–/25.3/28.6	–	–/28.2/33.9	–/31.3/36.4	–	–/34.5/42.1
IMP‡	**54.6/56.9/57.0**	36.2/37.7/37.8	15.5/19.5/22.4	15.9/20.2/24.5	15.9/20.2/24.2	19.8/25.0/28.6	20.4/26.3/32.1	20.5/26.1/31.6
KERN‡	**54.6**/56.8/56.9	33.2/34.6/34.6	18.1/23.8/26.8	18.7/26.2/32.0	18.7/26.2/31.9	22.7/29.4/33.1	23.5/32.6/39.4	23.5/32.5/39.5
Motif‡	54.0/56.6/56.7	35.4/37.1/37.2	18.0/23.5/27.6	18.5/25.7/30.7	18.5/25.7/30.5	23.1/31.0/35.9	23.6/32.9/39.2	23.6/32.8/38.9
VCTree‡	53.9/56.2/56.3	35.5/37.2/37.2	18.7/23.9/27.5	18.6/26.0/31.4	18.6/25.7/30.9	24.4/31.0/35.6	24.4/33.2/39.9	24.4/32.8/39.3
TGIR-B	54.3/56.6/56.7	**36.4/38.0/38.1**	19.2/24.6/28.3	19.8/27.3/33.4	19.8/27.2/33.1	**24.5/32.5/36.6**	25.0/34.6/42.0	25.0/34.5/41.7
TGIR-G	54.2/56.6/56.7	36.0/37.5/37.5	**19.7/25.5/29.1**	**20.7/28.7/34.2**	**20.7/28.7/34.2**	**24.5**/31.8/36.3	**25.6/35.4/42.1**	**25.6/35.4/42.1**

The best performances are shown in bold
The ‡ stands for our re-implemented results
k stands for the number of predicates that is allowed to predicted for each related pair
The three numbers in each cell are the results of R@20/50/100

comes at the expense of performance drop on R@K. Our TGIR method dominantly surpasses most methods especially on mR@K with mGAT, except that the Seq2Seq employs reinforcement learning technique which is not the main point of our work.

Among TYPE-I methods, Motif and VCTree are the most relevant to ours. TGIR using multi-branch tree structure outperforms Motif and yields comparable performance with VCTree which uses a binary tree structure. We observe that TGIR especially achieves better performances on mR@K, which should be attributed to the HCP_L sub-module as described in ablation study. Besides, TGIR and VCTree are the top-2 methods in terms of mR@K. It indicates that hierarchical structure is superior to plain one in terms of contextual information modeling. Comparing TGIR with VCTree, the tree structure of VCTree is constructed by gradually selecting the related pair with the maximum learnable score, which has less interpretability. Our HET is heuristically generated and the structure stands for the subordination and juxtaposition among the objects. Further, TGIR contains two contextual propagation paths along HET so that those relatively strong and key relationships included in these two paths are especially emphasised, which results in better mR@K.

The VRD dataset is originally for visual relationship detection task, which is very similar to scene graph generation. Therefore, we make a comparison on VRD dataset as shown in Table 4. Similarly, we report results of the previous methods for VRD task in the upper part and the results of our re-implemented mainstream scene graph generation methods in the bottom part. The models in the bottom part and RelDN Zhang et al. (2019) use the object detector pre-trained on MS-COCO and finetuned on VRD. Zoom-Net Yin et al. (2018) states that they use ImageNet pre-trained weights and others remain unknown. It is shown that our method yields better results compared to the state-of-the-arts.

4.3.2 Comparison on Key Relationship Prediction

When it comes to key relationship prediction, we mainly compare our methods with Motif and VCTree on VG-KR under the PREDCLS protocol to eliminate the interference of errors from the object detector. We conduct three groups of experiments under three settings: (1) "w/o Sup": Models are trained on VG200, and directly evaluated on VG-KR. RRM is not applied. (2) A-Sup: Models are trained on VG200 using A-Sup scheme, and evaluated on VG-KR. RRM is applied. (3) D-Sup: Models are trained using D-Sup scheme, and evaluated on VG-KR. RRM is applied. As key relationships are much less than normal relationships according to Fig. 10a, we set the K in R@K and mR@K to small values, i.e., 5, 10, and 20. The results are shown in Table 5.

It is observed that TGIR dominantly outperforms other two competitors under the three settings. TGIR especially obtains better performance on R@K under the tuple-match rule, about 2% higher than other two methods, suggesting that the HET provides hints for estimating the importance of relationships and capturing humans' perceptive habits. TGIR-B performs well on R@K under the triplet-match rule while TGIR-G works better on mR@K under the triplet-match rule and R@K under the tuple-match rule. Comparing the three settings, models trained with D-Sup scheme provide an upper bound. We find that A-Sup scheme effectively improves performance compared to the setting of no supervision ("w/o Sup") and the improvement is especially significant on R@K under the tuple-match rule. It suggests that the approximate supervision indeed helps mining the important related pairs. Finally, RRM works well for all of these methods, which show its excellent transferability.

Qualitative Analysis We demonstrate some qualitative results in Fig. 14. Despite some small deviations, HET is mainly well constructed and close to human's analyzing process. We compare top five relationships predicted by TGIR with or without importance relationship supervision. It is shown that the top five relationships predicted by the model with importance relationship supervision are mainly between the top objects in HET. It verifies the rationality of HET again. These relationships focus on describing the major events of the pictures rather than the details.

Failure Cases Analysis We demonstrate some failure cases in Fig. 15. In order to get rid of the interference from the object detection, the scene graphs are predicted under the PREDCLS protocol. These failure cases can be categorized into two types: (1) The HET is not precisely constructed, for example, in the first case, the *engine_1* is wrongly governed by *man_1*, whose correct parent node should be *bike_1*. Despite this error, the HCP_T sub-module still successfully discovers the strong association between the *engine_1* and the *bike_1*, and correctly predicts the relationship ⟨*engine_1*, *on*, *bike_1*⟩, although this relationship may not be important. (2) Sometimes the objects in the image are too dense. As shown in the second case, since the *boat_3* is closed to *boat_1*, the model wrongly configures the related pairs, mistaking ⟨*man_1*, *on*, *boat_1*⟩ for ⟨*man_1*, *on*, *boat_3*⟩.

5 Experiments on Subsequent Applications

In this section, we attempt to conduct some experiments on subsequent applications to show the value of key relationship. As the scene graph has been widely adopted in image captioning and cross-modal retrieval, experiments are mainly conducted on these two applications. Besides, we will validate that the key relationship keeps the major content and helps restore the original image, through the image generation experiment.

Table 5 Performances (%) on VG-KR

	w/o Sup			A-Sup			D-Sup		
	Tri. R@K	Tri. mR@K	Tup. R@K	Tri. R@K	Tri. mR@K	Tup. R@K	Tri. R@K	Tri. mR@K	Tup. R@K
Motif‡	31.1/36.1/45.7	3.4/5.4/7.6	46.7/60.8/81.7	32.5/37.9/45.9	3.6/5.5/7.7	49.0/62.7/82.2	34.2/41.3/47.6	6.3/8.4/10.1	63.1/78.3/88.9
VCTree‡	31.3/37.0/46.5	3.9/6.2/9.1	46.9/61.9/82.0	32.4/38.0/46.8	4.2/6.4/9.1	50.3/63.1/82.8	35.5/41.7/47.9	6.9/8.7/10.3	63.9/78.1/88.8
TGIR-B	**32.0/38.2/47.5**	3.9/6.2/8.9	47.8/62.6/82.9	**33.5/39.0/47.6**	4.2/6.4/9.0	**51.4/64.6**/83.4	**36.3/43.5/49.2**	7.1/9.0/11.0	64.9/78.7/90.1
TGIR-G	31.8/37.8/47.1	**4.8/7.8/11.0**	**48.0/63.0/83.8**	32.9/38.4/47.4	**5.0/7.9/11.1**	51.1/**64.6/84.2**	35.8/42.9/48.0	**8.2/10.5/12.1**	**65.9/80.1/90.6**

The best performances are shown in bold
The ‡ stands for our re-implemented results
"Tri. R" and "Tri. mR" stand for R@K and mR@K under triplet-match rule
"Tup. R" stands for R@K under the tuple-match rule
K is set to 5, 10, and 20

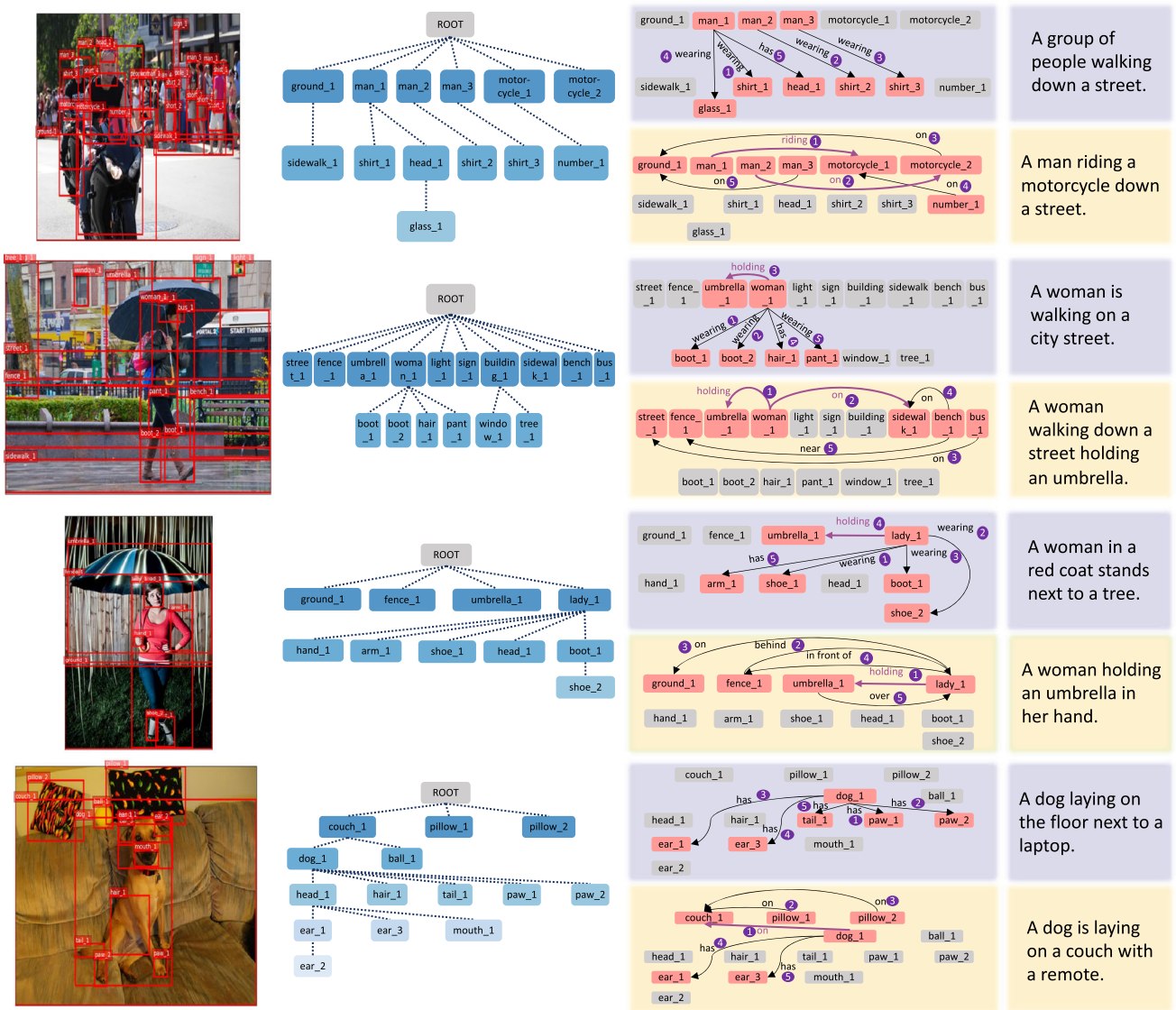


Fig. 14 Qualitative results. Four cases are shown. For each case, we demonstrate the picture with the bounding boxes, its corresponding HET, the top five relationships attached to the HET predicted by TGIR with or without importance relationship supervision (the bottom yellow block and the upper purple block) respectively. In these relationships, the purple arrows are key relationships matched with ground truth, and the purple numeric tags next to the relationships are their rankings, e.g., the "1" means that the relationship gets the highest score. Besides, we show the image captions generated using top-2 relationships

Table 6 Performances (%) on image captioning using different numbers of top relationships as input

Top n	Model	B@1	B@2	B@3	B@4	ROUGE-L	CIDEr	SPICE	Avg. growth
all	GCN-LSTM	72.0	54.7	40.5	30.0	52.9	91.1	18.1	
20	TGIR-B (Freq)	73.1	55.7	41.0	30.1	53.5	94.0	18.8	
	TGIR-B	74.9	**58.4**	**43.9**	**32.8**	54.9	101.7	19.8	0.06
	TGIR-B (D-Sup)	**75.0**	58.2	43.7	32.7	**55.1**	**102.2**	**19.9**	
5	TGIR-B (Freq)	70.7	53.2	38.6	28.0	51.7	84.4	17.2	
	TGIR-B	72.5	55.4	41.2	30.5	53.1	92.6	18.5	1.57
	TGIR-B (D-Sup)	**73.7**	**56.7**	**42.3**	**31.5**	**54.0**	**97.5**	**19.1**	
2	TGIR-B (Freq)	68.1	50.8	36.8	26.5	50.2	76.5	15.5	
	TGIR-B	70.8	53.4	39.2	28.7	51.8	86.4	17.6	2.10
	TGIR-B (D-Sup)	**72.3**	**55.2**	**41.0**	**30.4**	**53.1**	**92.2**	**18.4**	

The best performances are shown in bold

Table 7 Performances (%) of different scene graph generation methods on image captioning

Supervision	Model	B@1	B@2	B@3	B@4	ROUGE-L	CIDEr	SPICE
–	Motif	72.7	55.9	40.8	29.4	53.2	99.0	18.4
	VCTree	72.9	57.7	41.6	30.3	54.2	100.1	18.5
	TGIR-B	74.9	58.4	43.9	32.8	54.9	101.7	19.8
	TGIR-G	**75.3**	**59.3**	**44.2**	**33.7**	**55.8**	**102.9**	**20.3**
D-Sup	Motif	73.1	56.3	41.2	29.6	53.4	100.8	18.6
	VCTree	73.3	57.8	41.6	30.2	54.7	101.6	18.8
	TGIR-B	75.0	58.2	43.7	32.7	55.1	102.2	19.9
	TGIR-G	**75.8**	**59.9**	**44.8**	**34.4**	**56.3**	**103.8**	**20.6**

The best performances are shown in bold
Top 20 relationships are used as input

5.1 Evaluation on Image Captioning

We exploit the key relationships for image captioning. The experiments are conducted on VG-KR since it contains caption annotation from MS-COCO. To generate image caption, we select different numbers of predicted top relationships and feed them into the LSTM decoder following GCN-LSTM Yao et al. (2018). We re-implement GCN-LSTM and evaluate it on VG-KR since it is one of the state-of-the-art methods and meets our requirements well.

GCN-LSTM conducts graph convolution on the scene graph and injects all relation-aware region-level features into a two-layer LSTM decoder. It uses a simple two-layer MLP classifier to predict the pairwise relationship, which acts as the front-end scene graph generator. For a fair comparison, we replace the scene graph generator with our TGIR. Besides, we feed the relationship features rather than region-level features into the LSTM decoder, considering that the relationships which convey the events in the image are more helpful for description generation.

We compare three variants of our method: (1) TGIR-B, (2) TGIR-B (Freq): the predicted relationships are artificially re-ranked according to their frequency in VG-KR, and (3) TGIR-B (D-Sup): the model is trained using D-Sup scheme. We feed top n relationships into the decoder to generate the image caption. All mainstream metrics for image captioning evaluation are adopted.

As shown in Table 6, TGIR-B (Freq) with top 20 relationships input, outperforms GCN-LSTM because GCN-LSTM conducts graph convolution using relationships as edges and uses relation-aware region-level features as input, which is not as effective as our design that the relationship features are directly fed into the decoder. After applying D-Sup scheme, there is consistent performance improvement on overall metrics. This improvement is more and more significant as the number of input top relationships reduces. It is reasonable since the impact centers at top relationships. It suggests that our method provides more essential content with as few relationships as possible, which contributes to efficiency improvement.

In Table 7, we compare the performances of different scene graph generation methods on image captioning. It's shown that no matter using or not using key relationship supervision, our methods surpass other scene graph generation methods.

In Fig. 14, we show the image captions generated using top-2 relationships as input. In the second sample, when the top-2 relationships are ⟨*woman*, *wearing*, *boot_1/boot_2*⟩,

Table 8 Performances on cross-modal retrieval

Model	Top-1				Top-5				Top-5-CON			
	R@1 (%)	R@5 (%)	R@10 (%)	Med	R@1 (%)	R@5 (%)	R@10 (%)	Med	R@1 (%)	R@5 (%)	R@10 (%)	Med
TGIR-B	8.2/6.9	26.9/24.2	39.8/35.2	16.0/20.0	13.2/6.8	33.6/21.8	46.8/34.0	12.0/23.0	25.2/21.7	54.7/48.4	70.2/64.4	4.0/6.0
TGIR-B (A-Sup)	10.1/7.8	27.7/27.0	40.9/37.7	15.0/18.0	15.2/7.3	37.8/23.2	51.2/35.7	10.0/21.0	25.5/23.2	57.2/50.9	71.8/68.0	4.0/5.0
TGIR-B (D-Sup)	**16.6/15.7**	**46.6/43.6**	**62.3/60.6**	**6.0/7.0**	**19.8/10.0**	**48.2/29.9**	**62.0/44.1**	**6.0/14.0**	**37.2/28.9**	**67.6/61.8**	**81.7/77.2**	**2.0/3.0**

The two values in each cell stand for results of the image-to-text and text-to-image retrieval respectively. Three settings are included: using top 1 relationship (Top-1), top 5 relationships (Top-5), and one sentence by connecting the top 5 relationships (Top-5-CON). For the median rank (Med), the lower the better
The best performances are shown in bold

Table 9 Performances of different methods on cross-modal retrieval

Sup	Model	Top-1				Top-5				Top-5-CON			
		R@1 (%)	R@5 (%)	R@10 (%)	Med	R@1 (%)	R@5 (%)	R@10 (%)	Med	R@1 (%)	R@5 (%)	R@10 (%)	Med
–	Motif	7.1/6.4	25.8/24.0	39.1/34.1	17.0/19.0	13.1/5.6	32.3/21.0	45.3/33.0	13.0/23.0	24.1/20.4	53.8/47.5	69.3/62.7	4.0/7.0
	VCTree	7.5/6.8	26.6/24.1	39.0/34.6	17.0/19.0	13.0/5.9	33.5/21.3	46.2/33.5	13.0/23.0	24.4/20.5	54.0/48.3	70.0/63.4	4.0/6.0
	TGIR-B	8.2/6.9	26.9/24.2	39.8/35.2	16.0/20.0	13.2/6.8	33.6/21.8	46.8/34.0	12.0/23.0	25.2/21.7	54.7/48.4	70.2/64.4	4.0/6.0
	TGIR-G	**10.6/8.1**	**27.9/26.5**	**41.3/37.8**	**16.0/19.0**	**15.8/7.3**	**33.7/21.8**	**48.5/34.0**	**11.0/22.0**	**28.2/23.5**	**58.0/52.3**	**71.4/68.1**	**4.0/5.0**
D-Sup	Motif	14.7/14.5	45.1/41.7	60.9/57.6	7.0/8.0	18.5/8.9	45.6/28.4	60.7/42.6	7.0/14.0	33.4/28.2	66.2/60.6	80.2/74.8	3.0/4.0
	VCTree	15.6/15.2	46.1/42.7	61.0/59.4	7.0/7.0	19.0/9.9	46.9/29.3	61.4/43.4	6.0/14.0	33.8/28.8	66.3/60.8	80.8/75.2	3.0/3.0
	TGIR-B	16.6/15.7	46.6/43.6	62.3/60.6	6.0/7.0	19.8./10.0	48.2/29.9	62.0/44.1	6.0/14.0	37.2/28.9	67.6/61.8	81.7/77.2	2.0/3.0
	TGIR-G	**18.6/16.9**	**48.1/44.3**	**63.6/61.8**	**6.0/6.0**	**20.6/10.4**	**49.3/30.2**	**63.7/45.0**	**6.0/13.0**	**38.6/29.1**	**68.0/62.5**	**82.6/78.2**	**2.0/3.0**

The best performances are shown in bold
The metric settings are the same as Table 8

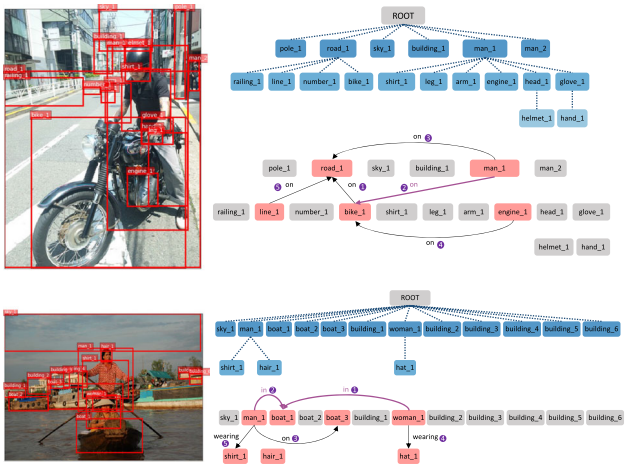


Fig. 15 Failure cases analysis. For each case, we demonstrate the picture with the bounding boxes, its corresponding HET, the top five relationships attached to the HET predicted by TGIR-B (D-Sup). The meanings of the purple arrows and numeric tags are the same as those in Fig. 14

the generated caption cannot capture the essential content that *the woman is holding an umbrella*. When the top-2 relationships contain ⟨*woman*, *holding*, *umbrella*⟩, the caption successfully reveals the major content. In some cases, we observe that the top-2 relationships are not totally consistent with the image caption. It may because of the bias problem in the literature of image captioning. Besides, as the input contains visual features, it is difficult to fully restricted information beyond the top-2 relationships. Despite this, it still suggests that key relationships are more helpful for generating a description that highly fits the major events in an image.

5.2 Evaluation on Cross-Modal Retrieval

As key relationships are claimed to be most relevant to the major events in an image, it is expected that they benefit the cross-modal retrieval. Specifically, we exploit the widely-used image-text matching model SCAN (Lee et al., 2018) to help compute the similarity between text and image. 1,000 images are randomly chosen from the test set of VG-KR as the image gallery, and the predicted top 1 or 5 relationships are collected as the text gallery, which are converted into short sentences in "subject-predicate-object" style. We design three settings for computing text-image similarity: (1) using top 1 relationship (Top-1), (2) using top 5 relationships (Top-5), and (3) using the single sentence by connecting the top 5 relationships (Top-5-CON). The recall (R@K, K is 1, 5, 10) and the median rank (Kim et al., 2019) of the correctly retrieved images or text are used as the metrics. We mainly use three models to generate the sorted relationships: (1) TGIR-B, (2) TGIR-B (A-Sup), which is trained using A-Sup scheme, and (3) TGIR-B (D-Sup), which is trained using D-Sup scheme. The results are shown in Table 8. It is shown that under all settings, the retrieval performance will be improved using more important relationships. It suggests that the key relationships are much more image-specific and therefore they are more practical for supporting other tasks.

Similar to Sect. 5.1, we compare different scene graph methods on cross-modal retrieval as shown in Table 9. TGIR methods performs better with or withour key relationship supervision.

In Fig. 16, we demonstrate some qualitative results. In these samples, the major events in the query image (left column) can be decomposed into several key relationships (as shown in the scene graph). The key relationships make it possible to designate the target content to be retrieved, e.g., to retrieve *man holding board* or *man standing on beach* in the third sample, while it is difficult to achieve this goal with those trivial relationships.

5.3 Qualitative Evaluation on Image Generation

Scene graph can be applied to image generation. Extracting the key relationships is helpful for generating the target image that meets the requirements.

We apply the sg2im method (Johnson et al., 2018) and use the top 5 relationships from TGIR-G (D-Sup) and TGIR-G respectively as input. The qualitative results are shown in Fig. 17. It is observed that the relationships predicted by TGIR-G (D-Sup) are more important and thus the generated images are roughly more consistent with the original ones, although it's difficult to generate a high-resolution complicated scene image. For example, in the first row, TGIR-G (D-Sup) successfully focuses on the main content, the building with a clock, while the car is secondary. In the third row, TGIR-G (D-Sup) notices that the main content is "elephant near elephant", and thus two elephants can be roughly observed from the generated image.

6 Conclusion

In this work we propose a new scene graph modeling formulation and make an attempt to push the study of scene graph generation towards practicability and rationalization. We generate a hierarchical scene graph inspired by humans' scene parsing procedure, and further prioritize the key relationships as far as possible. Experimental results show that the reasonable constructed hierarchical scene structure contributes to scene graph inference and boosts the key relationship prediction. It is found that some objective cues are effective for estimating the relationship importance. Besides, experiments on image captioning and cross-modal retrieval

Fig. 16 The text-to-image retrieval results using some key relationships in the scene graph

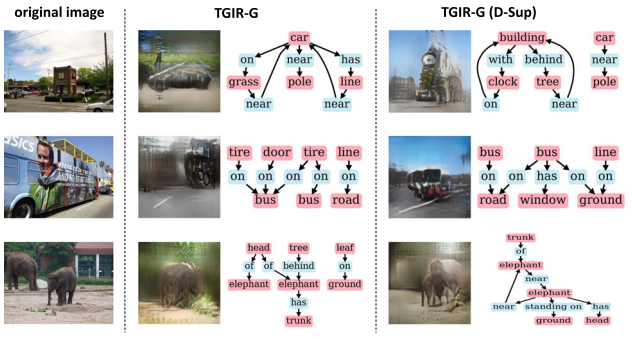


Fig. 17 Image generation results using top 5 relationships from TGIR and TGIR-G (D-Sup) model

suggest that key relationships are not just for appreciating, but indeed have great potential to support other applications.

Generally, hierarchical scene graph generation has two additional advantages. As illustrated in Sect. 4.2.1, the association between the levels of HET and relationship importance makes it possible to reasonably and conveniently adjust the scale of a scene graph. When serving for subsequent applications, the scale of a scene graph usually should be restricted. The current common practice is to keep the relationships with the highest triplet scores without thinking their correspondence with the main content of the image. A more reasonable practice is to keep the key relationships. Furthermore, during the scene graph inference stage, conventional practice requires exhaustive relationship prediction for every pair of objects. Our method has potential to take a further step towards efficient inference, getting rid of the $O(N^2)$ complexity (Li et al., 2018) of the conventional practice, i.e., only the relationships between a parent and its children nodes, and any two sibling nodes are predicted. We intend to improve the construction of HET in our future work so that more efficient inference is possible while the accuracy is still guaranteed.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s11263-023-01817-7.

Acknowledgements This work is partially supported by Natural Science Foundation of China under contracts Nos. U21B2025, U19B2036, 61922080, and National Key R&D Program of China No. 2021ZD0111901.

References

Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Lawrence Zitnick, C., & Parikh, D. (2015). Vqa: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 2425–2433).

Biederman, I. (2017). On the semantics of a glance at a scene. In: *Perceptual Organization* (pp. 213–253). Routledge.

Bordalo, P., Gennaioli, N., & Shleifer, A. (2012). Salience theory of choice under risk. *The Quarterly Journal of Economics, 127*(3), 1243–1285.

Chen, S., Jin, Q., Wang, P., & Wu, Q. (2020). Say as you wish: Fine-grained control of image caption generation with abstract scene graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 9962–9971).

Chen, T., Yu, W., Chen, R., & Lin, L. (2019). Knowledge-embedded routing network for scene graph generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 6163–6171).

Chiou, M.J., Ding, H., Yan, H., Wang, C., Zimmermann, R.,& Feng, J. (2021). Recovering the unbiased scene graphs from the biased ones. In *Proceedings of the ACM International Conference on Multimedia (ACM-MM)* (pp. 1581–1590).

Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. In: *Advances in Neural Information Processing Systems (NIPS) Workshop on Deep Learning*.

Dai, B., Zhang, Y., & Lin, D. (2017). Detecting visual relationships with deep relational networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3298–3308).

Deng, Z., Hu, X., Zhu, L., Xu, X., Qin, J., Han, G., & Heng, P.A. (2018). R3net: Recurrent residual refinement network for saliency detection. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (AAAI)* (pp. 684–690).

Dhamo, H., Farshad, A., Laina, I., Navab, N., Hager, G.D., Tombari, F.,& Rupprecht, C. (2020). Semantic image manipulation using scene graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 5213–5222).

Gu, J., Joty, S., Cai, J., Zhao, H., Yang, X.,& Wang, G. (2019). Unpaired image captioning via scene graph alignments. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 10,323–10,332).

Gu, J., Zhao, H., Lin, Z., Li, S., Cai, J., & Ling, M. (2019). Scene graph generation with external knowledge and image reconstruction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1969–1978).

Guo, Y., Gao, L., Wang, X., Hu, Y., Xu, X., Lu, X., Shen, H.T., & Song, J. (2021). From general to specific: Informative scene graph generation via balance adjustment. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 16,383–16,392).

Han, F., & Zhu, S. C. (2008). Bottom-up/top-down image parsing with attribute grammar. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 31*(1), 59–73.

He, S., Tavakoli, H.R., Borji, A.,& Pugeault, N. (2019). Human attention in image captioning: Dataset and analysis. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 8529–8538).

Herzig, R., Bar, A., Xu, H., Chechik, G., Darrell, T., & Globerson, A. (2020). Learning canonical representations for scene graph to image generation. In *Proceedings of European Conference on Computer Vision (ECCV)*, vol. 12371, (pp. 210–227). Springer.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9*(8), 1735–1780.

Hou, Q., Cheng, M.M., Hu, X., Borji, A., Tu, Z., & Torr, P. (2017). Deeply supervised salient object detection with short connections. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3203–3212).

Itti, L., Koch, C., & Niebur, E. (1998). A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 20*(11), 1254–1259.

Johnson, J., Gupta, A., & Fei-Fei, L. (2018). Image generation from scene graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 1219–1228).

Johnson, J., Krishna, R., Stark, M., Li, L.J., Shamma, D., Bernstein, M., & Fei-Fei, L. (2015). Image retrieval using scene graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3668–3678).

Kahneman, D., Slovic, S.P., Slovic, P.,& Tversky, A. (1982). Judgment under uncertainty: Heuristics and biases. Cambridge University Press.

Kim, D.J., Choi, J., Oh, T.H.,& Kweon, I.S. (2019). Dense relational captioning: Triple-stream networks for relationship-based captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 6271–6280).

Klein, D.A., & Frintrop, S. (2011). Center-surround divergence of feature statistics for salient object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 2214–2219).

Koner, R., Sinhamahapatra, P.,& Tresp, V. (2020). Relation transformer network. arXiv:2004.06193.

Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., Chen, S., Kalantidis, Y., Li, L. J., Shamma, D. A., Bernstein, M. S., & Fei-Fei, L. (2017). Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision (IJCV), 123*(1), 32–73.

Lee, K.H., Chen, X., Hua, G., Hu, H., & He, X. (2018). Stacked cross attention for image-text matching. In *Proceedings of European Conference on Computer Vision (ECCV)* vol. 11208, (pp. 201–216). Springer.

Li, G., & Yu, Y. (2015). Visual saliency based on multiscale deep features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 5455–5463).

Li, R., Zhang, S., Wan, B.,& He, X. (2021). Bipartite graph network with adaptive message passing for unbiased scene graph generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 11,109–11,119).

Li, X., & Jiang, S. (2019). Know more say less: Image captioning based on scene graphs. *IEEE Transactions on Multimedia (TMM), 21*(8), 2117–2130.

Li, Y., Ouyang, W., Wang, X., & Tang, X. (2017). Vip-cnn: Visual phrase guided convolutional neural network. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 7244–7253).

Li, Y., Ouyang, W., Zhou, B., Shi, J., Zhang, C., & Wang, X. (2018) Factorizable net: an efficient subgraph-based framework for scene graph generation. In *Proceedings of European Conference on Computer Vision (ECCV)* vol. 11205, (pp. 346–363). Springer.

Li, Y., Ouyang, W., Zhou, B., Wang, K., & Wang, X. (2017). Scene graph generation from objects, phrases and region captions. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 1261–1270).

Liang, X., Lee, L., Xing, E.P. (2017). Deep variation-structured reinforcement learning for visual relationship and attribute detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 4408–4417).

Liang, Y., Bai, Y., Zhang, W., Qian, X., Zhu, L., & Mei, T. (2019). Vrr-vg: Refocusing visually-relevant relationships. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 10,403–10,412).

Lin, L., Wang, G., Zhang, R., Zhang, R., Liang, X., & Zuo, W. (2016). Deep structured scene parsing by learning with image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 2276–2284).

Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C.L. (2014). Microsoft coco: Common objects in context. In *Proceedings of European Conference on Computer Vision (ECCV)* vol. 8693, (pp. 740–755). Springer.

Lin, X., Ding, C., Zeng, J.,& Tao, D. (2020). Gps-net: Graph property sensing network for scene graph generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3746–3755).

Liu, N., Han, J., & Yang, M.H. (2018). Picanet: Learning pixel-wise contextual attention for saliency detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3089–3098).

Lu, C., Krishna, R., Bernstein, M.,& Fei-Fei, L. (2016). Visual relationship detection with language priors. In: Proceedings of European Conference on Computer Vision (ECCV) vol. 9905, (pp. 852–869). Springer.

Lu, Y., Rai, H., Chang, J., Knyazev, B., Yu, G., Shekhar, S., Taylor, G.W., & Volkovs, M. (2021). Context-aware scene graph generation with seq2seq transformers. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 15,931–15,941).

Lv, J., Xiao, Q., & Zhong, J. (2020). Avr: Attention based salient visual relationship detection. arXiv:2003.07012.

Miller, G. A. (1992). Wordnet: A lexical database for English. *Communication of the ACM, 38*(11), 39–41.

Navon, D. (1977). Forest before trees: The precedence of global features in visual perception. *Cognitive Psychology, 9*(3), 353–383.

Nguyen, K., Tripathi, S., Du, B., Guha, T., & Nguyen, T.Q. (2021) In defense of scene graphs for image captioning. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 1407–1416).

Pennington, J., Socher, R., & Manning, C. (2014). Glove: Global vectors for word representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 1532–1543).

Peyre, J., Laptev, I., Schmid, C., & Sivic, J. (2017). Weakly-supervised learning of visual relations. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 5179–5188).

Pont-Tuset, J., Uijlings, J., Changpinyo, S., Soricut, R., & Ferrari, V. (2020). Connecting vision and language with localized narratives. In *Proceedings of European Conference on Computer Vision (ECCV)* vol. 12350, (pp. 647–664). Springer.

Qi, M., Li, W., Yang, Z., Wang, Y.,& Luo, J. (2019). Attentive relational networks for mapping images to scene graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3957–3966).

Ren, S., He, K., Girshick, R.,& Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS)* (pp. 91–99).

Schuster, S., Krishna, R., Chang, A., Fei-Fei, L.,& Manning, C.D. (2015) Generating semantically precise scene graphs from textual descriptions for improved image retrieval. In *Proceedings of the Fourth Workshop on Vision and Language* (pp. 70–80).

Sharma, A., Tuzel, O., & Jacobs, D.W. (2015). Deep hierarchical parsing for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 530–538).

Shi, J., Zhang, H., & Li, J. (2019). Explainable and explicit visual reasoning over scene graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 8376–8384).

Socher, R., Lin, C.C., Manning, C.,& Ng, A.Y. (2011). Parsing natural scenes and natural language with recursive neural networks. In *Proceedings of the International Conference on Machine Learning (ICML)* (pp. 129–136).

Suhail, M., Mittal, A., Siddiquie, B., Broaddus, C., Eledath, J., Medioni, G., Sigal, L. (2021). Energy-based learning for scene graph gener-

ation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 13,936–13,945).
Tai, K.S., Socher, R., Manning, C.D. (2015). Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)* (pp. 1556–1566).
Tang, C., Xie, L., Zhang, X., Hu, X., Tian, Q. (2022). Visual recognition by request. arXiv:2207.14227.
Tang, K., Niu, Y., Huang, J., Shi, J., Zhang, H. (2020). Unbiased scene graph generation from biased training. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3716–3725).
Tang, K., Zhang, H., Wu, B., Luo, W., Liu, W. (2019). Learning to compose dynamic tree structures for visual contexts. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 6619–6628).
Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. In *Advances in Neural Information Processing Systems (NIPS)* (pp. 5998–6008).
Velickovic, P., Cucurull, G., Casanova, A., Romero, A., Liò, P.,& Bengio, Y. (2018). Graph attention networks. In *International Conference on Learning Representations (ICLR)*.
Wang, L., Lu, H., Ruan, X., & Yang, M.H. (2015). Deep networks for saliency detection via local estimation and global search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3183–3192).
Wang, S., Wang, R., Yao, Z., Shan, S., & Chen, X. (2020). Cross-modal scene graph matching for relationship-aware image-text retrieval. In *Proceedings of the IEEE Winter Conference on Applications of Computer Vision (WACV)* (pp. 1508–1517).
Wang, T., Borji, A., Zhang, L., Zhang, P., & Lu, H. (2017). A stagewise refinement model for detecting salient objects in images. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 4019–4028).
Wang, W., Wang, R.,& Chen, X. (2021). Topic scene graph generation by attention distillation from caption. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 15,900–15,910).
Wang, W., Wang, R., Shan, S., & Chen, X. (2019). Exploring context and visual pattern of relationship for scene graph generation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 8188–8197).
Wang, W., Wang, R., Shan, S., & Chen, X. (2020). Sketching image gist: Human-mimetic hierarchical scene graph generation. In *Proceedings of European Conference on Computer Vision (ECCV)* vol. 12358, (pp. 222–239). Springer.
Xie, S., Girshick, R., Dollár, P., Tu, Z.,& He, K. (2017). Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 5987–5995).
Xie, Y., Lu, H., & Yang, M. H. (2012). Bayesian saliency via low and mid level cues. *IEEE Transactions on Image Processing (TIP), 22*(5), 1689–1698.
Xu, D., Zhu, Y., Choy, C.B., & Fei-Fei, L. (2017). Scene graph generation by iterative message passing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 5410–5419).
Xu, N., Liu, A. A., Liu, J., Nie, W., & Su, Y. (2019). Scene graph captioner: Image captioning based on structural visual representation. *Journal of Visual Communication and Image Representation, 58*, 477–485.
Yan, S., Shen, C., Jin, Z., Huang, J., Jiang, R., Chen, Y.,& Hua, X.S. (2020). Pcpl: Predicate-correlation perception learning for unbiased scene graph generation. In *Proceedings of the ACM International Conference on Multimedia (ACM-MM)* (pp. 265–273).
Yang, J., Lu, J., Lee, S., Batra, D., Parikh, D. (2018). Graph r-cnn for scene graph generation. In *Proceedings of European Conference on Computer Vision (ECCV)* vol. 11205, (pp. 690–706). Springer.
Yang, X., Tang, K., Zhang, H.,& Cai, J. (2019). Auto-encoding scene graphs for image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 10,685–10,694).
Yao, T., Pan, Y., Li, Y.,& Mei, T. (2018). Exploring visual relationship for image captioning. In *Proceedings of European Conference on Computer Vision (ECCV)* vol. 11218, (pp. 711–727). Springer.
Yao, T., Pan, Y., Li, Y., & Mei, T. (2019). Hierarchy parsing for image captioning. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 2621–2629).
Yin, G., Sheng, L., Liu, B., Yu, N., Wang, X., Shao, J.,& Loy, C.C. (2018). Zoom-net: Mining deep feature interactions for visual relationship recognition. In *Proceedings of European Conference on Computer Vision (ECCV)* vol. 11207, (pp. 330–347). Springer.
Yu, J., Chai, Y., Wang, Y., Hu, Y., & Wu, Q. (2021). Cogtree: Cognition tree loss for unbiased scene graph generation. In *Proceedings of International Joint Conferences on Artificial Intelligence (IJCAI)* (pp. 1274–1280).
Yu, R., Li, A., Morariu, V.I., & Davis, L.S. (2017). Visual relationship detection with internal and external linguistic knowledge distillation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 1974–1982).
Zareian, A., Karaman, S., & Chang, S.F. (2020a). Bridging knowledge graphs to generate scene graphs. In *Proceedings of European Conference on Computer Vision (ECCV)* vol. 12368, (pp. 606–623). Springer.
Zareian, A., Karaman, S.,& Chang, S.F. (2020b). Weakly supervised visual semantic parsing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 3736–3745).
Zareian, A., You, H., Wang, Z.,& Chang, S.F. (2020c). Learning visual commonsense for robust scene graph generation. In *Proceedings of European Conference on Computer Vision (ECCV)* vol. 12368, (pp. 642–657). Springer.
Zellers, R., Yatskar, M., Thomson, S., & Choi, Y. (2018). Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 5831–5840).
Zhang, H., Kyaw, Z., Chang, S.F., & Chua, T.S. (2017a) Visual translation embedding network for visual relation detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* pp. 5532–5540).
Zhang, H., Kyaw, Z., Yu, J., & Chang, S.F. (2017b). Ppr-fcn: Weakly supervised visual relation detection via parallel pairwise r-fcn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)* (pp. 4233–4241).
Zhang, J., Kalantidis, Y., Rohrbach, M., Paluri, M., Elgammal, A.,& Elhoseiny, M. (2019). Large-scale visual relationship understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)* (pp. 9185–9194).

Zhang, J., Shih, K.J., Elgammal, A., Tao, A., & Catanzaro, B. (2019). Graphical contrastive losses for scene graph parsing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 11,535–11,543).

Zhang, L., Zhang, J., Lin, Z., Lu, H., & He, Y. (2019).Capsal: Leveraging captioning to boost semantics for salient object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* pp. 6024–6033).

Zhong, Y., Wang, L., Chen, J., Yu, D., & Li, Y.(2020). Comprehensive image captioning via scene graph decomposition. In *Proceedings of European Conference on Computer Vision (ECCV)* vol. 12359, (pp. 211–229). Springer.

Zhu, L., Chen, Y., Lin, Y., Lin, C., & Yuille, A. (2011). Recursive segmentation and recognition templates for image parsing. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 34*(2), 359–371.