Grounding Scene Graphs on Natural Images via Visio-Lingual Message Passing

Aditay Tripathi[1] Anand Mishra[2] Anirban Chakraborty[1]
[1]Indian Institute of Science [2] Indian Institute of Technology Jodhpur
{aditayt,anirban}@iisc.ac.in mishra@iitj.ac.in
https://iiscaditaytripathi.github.io/sgl/

Abstract

This paper presents a framework for jointly grounding objects that follow certain semantic relationship constraints given in a scene graph. A typical natural scene contains several objects, often exhibiting visual relationships of varied complexities between them. These inter-object relationships provide strong contextual cues towards improving grounding performance compared to a traditional object query-only-based localization task. A scene graph is an efficient and structured way to represent all the objects and their semantic relationships in the image. In an attempt towards bridging these two modalities representing scenes and utilizing contextual information for improving object localization, we rigorously study the problem of grounding scene graphs on natural images. To this end, we propose a novel graph neural network-based approach referred to as Visio-Lingual Message PAssing Graph Neural Network (VL-MPAG Net). In VL-MPAG Net, we first construct a directed graph with object proposals as nodes and an edge between a pair of nodes representing a plausible relation between them. Then a three-step inter-graph and intra-graph message passing is performed to learn the context-dependent representation of the proposals and query objects. These object representations are used to score the proposals to generate object localization. The proposed method significantly outperforms the baselines on four public datasets.

1. Introduction

> "What are the mental events that transpire when our eyes alight upon a novel scene? The comprehension that is achieved is not a simple listing of the creatures and objects. Instead, our mental representation includes a specification of the various relations that exist among these entities."
>
> –*Biederman et al., [1]*

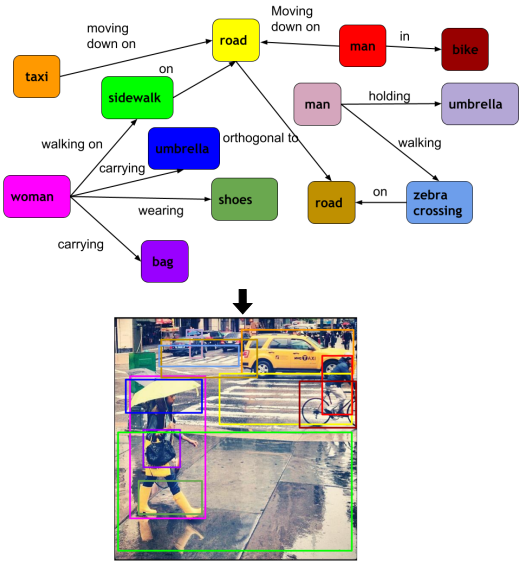

Figure 1: **Our goal: Grounding scene graph on image.** Given a scene graph and an image, we ground (or localize) objects and, thereby, indirectly visual relationships as well jointly on the image. **[Best viewed in color].**

The linking of *concepts to context* is referred to as 'grounding' [3]. In visual grounding, natural scene is the context, whereas concepts may be expressed using different modalities of queries such as sketch [32], natural image [13], speech [4], text [17, 22, 41] or scene graph [16]. In many computer vision tasks such as image generation [21] and image editing [31], scene graphs have been a popular choice as a query owing to their capability of expressing complex scenes having multiple object instances and semantic relationships among them in a concise, non-ambiguous, and structured way. Further, as noted in [28], "scene graphs explicitly provide a scene's geometry, topology, and semantics, making them compelling representations for navigation". In fact, scene graphs have proven their utility in embodied AI [40], where the scene prior is often encoded as a scene graph, and grounding them in an

environment helps embodied agents navigate efficiently.

Foresighting the aforesaid applications, Johnson et al. [16] have introduced the task of scene graph grounding as an auxiliary task to scene graph-based image retrieval as an early work in this direction. While using the graphs as queries for grounding objects is, in principle, exciting, one natural question is how to construct such queries. One possible direction is to use natural language sentences to graph generation [30], which is yet to achieve an acceptable level of performance for long and complex sentences and is an open area of research [36]. Another possibility is to use a carefully-designed user interface where non-expert users can quickly draw graphs of arbitrary complexity [16, 43]. Further, one can also obtain scene graph queries by choosing them from a fixed set of scene graphs representing the spatial configuration of objects in a scene obtained using commonsense knowledge [11] such as $\{\langle$ Monitor, on, Table$\rangle$, $\langle$ Keyboard, near, Monitor$\rangle$, $\langle$Chair, next to, Table$\rangle$, $\langle$Person, sitting on, Chair$\rangle\}$. Regardless of the method used to obtain scene graph queries, our scope in this paper is to study scene graph grounding as a standalone task.

In this work, we formulate and study the task of grounding scene graphs on images as defined in Figure 1, in a principled manner. Further, we propose a novel, robust and effective solution strategy suitable for the query data structure and the task at hand; and provide rigorous experiments and analysis on large-scale computer vision benchmarks. We hope this work will help establish the scene graph grounding as an important and stand-alone cross-modal computer vision problem, thereby leading to exciting contributions toward this open problem.

In this work, we propose a novel method to solve the scene graph grounding problem. To this end, given a query scene graph G^l and an image, we first obtain object proposals on the image using a region proposal network and construct a proposal graph G^v. Note that the proposal graph contains object proposals and trainable visual relationship embeddings as representations of nodes and edges, respectively. Now, suppose graph G^l and G^v contain m and n nodes respectively, then we add $m \times n$ auxiliary directed edges between nodes of G^l and G^v to construct a composite visio-lingual graph G^{vl}. These auxiliary directed edges allow us to learn the proposal representations relevant to the query graph. We then perform message-passing operations on G^{vl} to learn the contextual proposal and object representation. These learned proposals are scored against each query object to perform visual grounding. We refer to our approach as a **V**isio-**L**ingual **M**essage **PA**ssing Graph Neural **Net**work or VL-MPAG Net in short.

We evaluate VL-MPAG Net on four public datasets, namely Visual Genome [20], VRD [25], COCO-stuff [2], and SG [16], and compare it against the following baselines: (i) *Node-only approach* where only the objects in the scene graph query are localized without leveraging the relationship constraints, (ii) an approach where *flattened triplets obtained from scene graph query* is utilized to perform grounding using a state-of-the-art approach [17]. (iii) *CRF-based approach* proposed by [16] where they build a conditional random field (CRF) over the bounding boxes on an image and perform maximum-a-posterior estimation for object localization. These approaches either do not leverage relationships or fail to exploit the structural information present in the graph and thus fall short in performance. Contrary to these approaches, VL-MPAG Net jointly grounds objects that follow certain semantic relationship constraints given in a scene graph and thereby, outperform them. The implementation for this work is provided at https://iiscaditaytripathi.github.io/sgl/.

Contributions: We make the following contributions: (i) We pose the scene-graph grounding as a standalone problem in a principled manner. (ii) We propose a novel model – VL-MPAG Net towards solving this problem. The VL-MPAG Net has two novel characteristics. Firstly, a *query-guided proposal graph generation* that utilizes the relationships in the query graph to generate a sparse proposal graph with relevant edges. Secondly, a *visio-lingual message passing network* that learns a query-conditioned structured representation for the object proposals and the query entities to generate better localization. (iii) We demonstrate efficacy of VL-MPAG Net via rigorous experiments, ablations, and analysis on modern large-scale public benchmarks.

2. Related works

Scene Graph in Computer Vision: A scene graph is a structured representation of a scene that can precisely and unambiguously represent multiple objects and their semantic relationships. Scene graphs play a pivotal role in holistic scene understanding and are a popular way to represent visual knowledge [20]. Being semantically rich in representation, scene graphs have shown their utility in many computer vision tasks such as visual question answering [9, 5], image retrieval [16, 35], natural scene generation [15, 44, 21], and high-level image editing [7, 31]. In this work, we study scene graphs for grounding multiple objects and relationships jointly on the image.

Query-guided Object Localization: In query-guided object localization, the concepts (or queries) that need to be localized on the image are expressed using various modalities in the literature. In [32] and [13], authors use the hand-drawn sketch and natural image of an object, respectively, to express the concept of a 'single object'. Cheng et al. [4] use speech input to localize and segment the concept of nouns, i.e., objects and adjectives. Interaction between a subject and an object has been represented by a visual relationship, i.e., $\langle$subject, predicate, object$\rangle$ triplet. The task where both subject and object constrained by a

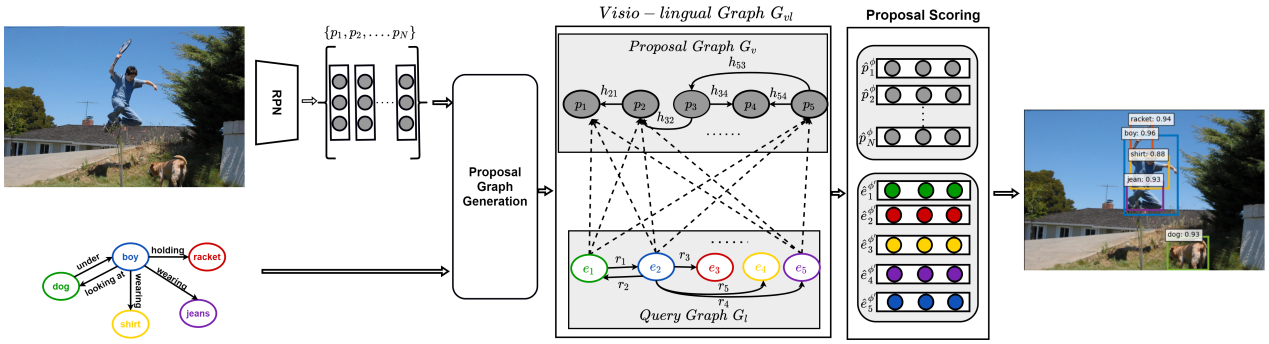

Figure 2: The proposed scene-graph grounding framework (VL-MPAG Net) works in the following steps: (i) **Proposal graph generation:** A proposal graph is first constructed using object proposals obtained from RPN as nodes (shown using gray nodes), and edges defined using the relations present in the query. Directed edges from the query nodes to the proposals nodes (shown using dotted arrows) are also included to connect the query and proposal graph (Section 3.2). (ii) **Structured graph learning:** Here, structured representation of proposals and queries are learned by a three-step message passing using edges from the query nodes to the proposals, and in the query and the proposal graphs independently (Section 3.3), and (iii) **Proposal scoring** the object proposals are finally scored against the query nodes to localize objects (Section 3.4).

visual relationship, need to be grounded on the image is referred to as Referring Relationship [25, 19, 12]. Authors in [26, 22, 17, 8, 45] use single-line sentence or short phrase as query to ground all the mentioned objects. The idea of scene graphs in computer vision literature has triggered encoding complex semantic concepts (such as the interaction between multiple object categories and instances) in a concise and structured form. Considering this, in a seminal work, Johnson et al. [16] utilized scene graph queries for localizing objects constrained by visual relationships and posed it as a maximum-a-posterior estimation problem. We take this work further by presenting a novel graph neural network-based approach and large-scale evaluation for grounding scene graphs on images.

Graph Neural Networks: The graph neural network (GNN) was proposed to learn the representation of the entities present in the graph. They have several variants such as graph attention networks [33], graph convolution networks [6, 18] and message-passing networks [10]. Among these, message-passing networks learn the representation for both the nodes and edges in the graph and seen application in many fields such as knowledge graph completion [34] , visual relationship detection [14], scene graph generation [38] and scene understanding [42]. Unlike current literature, we proposed a novel visio- lingual message passing network to learn the structured representation for the heterogeneous multi-modal graphs. Graph neural network, in general, has been widely used for various scene graph-related tasks. Yang et al. [39] proposed graph R-CNN for the scene graph generation. They propose an attention-graph convolution network where the global context in the scene graph is used to update object nodes and relationship edge labels. Authors in [24] use a graph similarity network for grounding *small phrases*. Unlike ours, they assume the availability of natural language query, use its embedding in their framework, and perform message passing independently on visual and lingual graphs. In [15], authors use graph convolution for scene graph-to-image generation; they compute a scene layout by predicting bounding boxes and segmentation masks for objects. Then, they transform the layout into an image with a refinement network. Query-driven proposal graph generation, message passing on multi-modal visio-lingual graphs, and learning context-dependent representation for the proposals and query objects are some highlights of our proposed GNN-based approach, which differentiate us from the scene graph-based GNN literature.

3. Visio-Lingual Message PAssing Graph neural NETwork (VL-MPAG Net)

3.1. Background and Problem Formulation

A scene graph is a structured representation of the scene containing object instances as its nodes and the relationship between the instances as its edges. Typically, scene graphs also contain object attributes represented as nodes in the graph. However, in the context of this paper, we drop attributes and our scene graphs contain only objects as nodes and relationships as edges. Formally, given a set of object entities $O = \{e_1, e_2, \cdots, e_n\}$ and a set of relations $R = \{r_1, r_2, \cdots, r_m\}$, a scene graph is defined as $s = (\mathcal{V}, \mathcal{E})$ such that $\mathcal{V} \subset O$ and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{R}' \times \mathcal{V}$ is a set of labeled edges, where $R' \subset R$.

Let $\{I_u, G_u^l\}_{u=1}^M$ be the M-pairs of natural images and corresponding scene graphs, respectively in a trainset of a dataset. Further, let O and R be the set of all object entities

Notation	Meaning
G^v, G^l, G^{vl}	Proposal graph, query graph, visio-lingual graph
I_u, G^l_u	u^{th} natural image and u^{th} scene graph
O, R	Set of object entities and set of relations
$\mathcal{R}_u$	Set of region proposals for image I_u
$\Phi_i, i \in \{1, 2, \ldots, 6\}$	Neural networks
$\mathbf{W}, \mathbf{W}_i, i \in \{1, \ldots, 4\}$	Trainable matrices
e_k, e_j	Nodes in query graph G^l
r_{kj}	Edge in query graph between nodes e_k and e_j
p_k, p_j	Nodes in proposal graph G^v
h_{kj}	Edge in proposal graph between nodes p_k and p_j
S_{kl}	Similarity score between proposal p_k and entity e_l
$ES_{kj,i}$	Similarity score between edge h_{kj} and relation r_i

Table 1: Notations used in this paper.

and relations present in the dataset. Each query scene graph G^l_u is a set of triples $\{(e_i, r_j, e_k)\}$ such that the object entities $e_i, e_k \in O$ and the relations $r_j \in R$. During inference, given an image I_u and a scene-graph query G^l_u, the task of *grounding scene graphs* on natural images involves localizing objects on the image that correspond to the entities, in the scene graph, that follow the constraints given by the corresponding relations in the scene graph.

The proposed end-to-end trainable framework is illustrated in Figure 2. It works in the following three stages: (i) proposal graph generation, (ii) structured graph learning, and (iii) joint proposal scoring which we describe next.

3.2. Proposal Graph Generation

Given an image I_u, we generate a set of region proposals $\mathcal{R}_u$ using a region proposal network (RPN) proposed in [29]. It is a neural network that generates a fixed set of region proposals represented using bounding box coordinates and confidence score of being a foreground, i.e., one of the object categories in the query graph. It should be noted here that RPN does not provide an exact object category label for the generated proposal. One plausible approach for grounding scene graphs could be to score these region proposals against the entity nodes present in the scene graph query to generate object localization. However, this approach does not utilize the structural information in the query graph or the target image and is unlikely to be very effective.

The query scene graph explicitly captures the structural information present among the objects and is represented as the edges that denote the relationships between objects. However, to incorporate the structural information (inter-object semantic relationship) present among different regions in the target image, we create a graph with region proposals as the nodes and the edges as the relationships constraining them. The RPN gives a set of region proposals, and we propose a 'query-driven' strategy to establish directed edges between pairs of proposals by leveraging the semantic relations present in the query scene graph. If the proposals are fully connected, the proposal graph would have $O(|\mathcal{R}_u|^2)$-edges. However, the number of actual connections is restricted by the plausible set of relationships constrained by the visual semantic association between the objects in the scene and is much smaller than $O(|\mathcal{R}_u|^2)$.

Given a pair of region proposals (p_k, p_j) and their corresponding bounding box coordinates (B_k, B_j), we first estimate the representation of the proposals as follows: $p^\phi_k = \Phi_1(p_k)$ and $p^\phi_j = \Phi_1(p_j)$, where $p^\phi_k, p^\phi_j \in \mathbb{R}^d$ and Φ_1 is a neural network. Note that p_k and p_j denote the image region bounded by B_k and B_j on image respectively. We then compute the representation of the edge between these two nodes as $h^\phi_{kj} = \mathbf{W}\left[p^\phi_k, p^\phi_j, \Phi_2\left([\gamma_{k,j}, \gamma_{k,kj}, \gamma_{j,kj}]\right)\right]$. Here, $\mathbf{W} \in \mathbb{R}^{3d \times d}$, $h^\phi_{kj} \in \mathbb{R}^d$ and Φ_2 is neural network. The γs are computed using bounding boxes B_k and B_j as follows: given a pair of bounding boxes B_k and B_j, we first construct a union rectangular box B_{kj} that tightly encloses B_k and B_j, and then we compute geometric features for each pair of boxes in $\{(B_k, B_j), (B_k, B_{kj}), (B_j, B_{kj})\}$. For example, the geometric features for the pair (B_k, B_j) are computed as follows:

$$\gamma_{k,j} = \left[\ln \frac{|x_k - x_j|}{w_k}, \ln \frac{|y_k - y_j|}{h_k}, \ln \frac{w_j}{w_k}, \ln \frac{h_j}{h_k}\right]^T, \quad (1)$$

where, (x_k, y_k, w_k, h_k) are the bounding box coordinates of the box B_k; and these features are generated for every pair of object proposals. Now, to retain only those edges for which the representations obtained using visual cues $h^\phi_{k,j}$ are well aligned to at least one of the relations present on the given query scene graph, we score each pair of proposals (p_k, p_j) against the relations present in the scene graph query as follows:

$$relSim_{kj} = \max\left(\Theta\left(h^\phi_{kj}, r_i\right)\right)_{i=1}^{K}, \quad (2)$$

where, $\{r_1, r_2, \ldots, r_K\}$ is the set of relations present in the query graph G^l_u and Θ is cosine similarity. Now, we add directed edge from the proposal p_k to p_j if the relationship similarity score $relSim_{kj}$ is above a predefined threshold. This process is repeated for each pair of proposals to generate a directed graph with p^ϕ_k as the node representations and h^ϕ_{kj} as the edge representations. Please note that all the mappings in this graph generation process, i.e. $(\mathbf{W}, \Phi_1, \Phi_2)$ are learnable and are trained in an end-to-end fashion.

3.3. Structured Graph Learning

Let G^l_u be the directed graph representing the u^{th} scene-graph query, and G^v_u be the corresponding proposal graph generated from image I_u as described in the the previous section (Section 3.2). The nodes in the query graph G^l_u represent the objects, and the edges represent the relationship between the object nodes. We use Glove [27] to get the initial representation of the entities and relations in G^l_u. The representation of nodes in both of these graphs is updated by passing messages from the neighbors. However,

if the proposal representations are updated independently of the query nodes, the same proposal representation will be learned for different query scene graphs. For example, consider an image containing a person wearing a hat and shoes, and two different queries *(person, wearing, hat)* and *(person, wearing, shoes)*. Concretely, in a proposal graph, a region proposal that corresponds to the query node *person* might have neighboring proposals that may correspond to *hat* or *shoes*. If we update the representation of proposal nodes independent of the query nodes, the representation of the proposal gets evenly influenced by all its neighbors, even though some of the neighbors do not correspond to any of the query nodes. To mitigate this problem, we add directed auxiliary edges from each node of the query graph G_u^l to each node of the proposal graph G_u^v as shown by the dotted edges in Figure 2 and construct a combined *visio-lingual graph*. A three-step message passing on this graph is performed to update the representation of nodes.

In the first step, message passing is performed on the auxiliary edges from the query graph to the proposal graph, and the representation of object proposals is updated as:

$$\bar{p}_k^{\phi} = \mathbf{W}_1 p_k^{\phi} + \mathbf{W}_2 \sum_j sim_{kj} \cdot e_j^{\phi'}, \tag{3}$$

$$sim_{kj} = \frac{e^{\left[\left(\mathbf{W}_3 p_k^{\phi}\right)^T \left(\mathbf{W}_4 e_j^{\phi'}\right)\right]}}{\sum_l e^{\left[\left(\mathbf{W}_3 p_k^{\phi}\right)^T \left(\mathbf{W}_4 e_l^{\phi'}\right)\right]}}, \tag{4}$$

where, $\mathbf{W}_1, \mathbf{W}_2, \mathbf{W}_3, \mathbf{W}_4 \in \mathbb{R}^{d \times d}$ and $e_j^{\phi'}$ is the representation of entity e_j in the query graph obtained using Glove. In this step, a weighted sum of the representation of the query nodes is added to each proposal's representation. Also, the weight depends on the compatibility of the proposal representation with the query nodes. This step helps incorporate query information in the proposal representation. In other words, for a target image, the final representation of the region proposals will be learned differently for each query.

In the second step, message passing is performed on the query scene graph. For the query graph G_u^l, the node representations are updated as follows:

$$\widehat{r}_{kj}^{\phi'} = \Phi_3\left(\left[e_k^{\phi'}, e_j^{\phi'}, r_{kj}^{\phi'}\right]\right), \tag{5}$$

$$\widehat{e}_k^{\phi'} = \frac{1}{|nbd(e_k)|} \sum_{nbd(e_k)} \Phi_4\left(\left[e_k^{\phi'}, \widehat{r}_{kj}^{\phi'}\right]\right), \tag{6}$$

where $nbd(e_k)$ is the set of neighbouring nodes of e_k in the graph G_u^l and Φ_3, Φ_4 are two-layer neural networks. Further, in the third step, given the proposal graph G_u^v, the representation of the proposal nodes are updated as follows:

$$\widehat{h}_{kj}^{\phi} = \Phi_5\left(\left[\bar{p}_k^{\phi}, \bar{p}_j^{\phi}, h_{kj}^{\phi}\right]\right), \tag{7}$$

$$\widehat{p}_k^{\phi} = \frac{1}{|nbd(p_k)|} \sum_{nbd(p_k)} \Phi_6\left(\left[\bar{p}_k^{\phi}, \widehat{h}_{kj}^{\phi}\right]\right). \tag{8}$$

Here, $nbd(p_k)$ is the set of neighbours of node representing proposal p_k in the proposal graph G_u^v, and Φ_5, Φ_6 are two-layer neural networks. After learning the contextual representation of nodes in both graphs, the proposal nodes are scored against the query nodes to ground the scene graph on the image. The representations learned after the second and third steps of message passing can be made more expressive by using two layers of GNN because it enables the model to utilize a 2-hop neighborhood context.

3.4. Joint Proposal Scoring

Once the representation of the query objects and the region proposals are updated, a scoring function Θ is used to score the region proposals with the query objects. Consider a proposal $p_k \in \mathcal{R}_u$ with a label variable y_k. During the training phase, y_k is assigned a class c_{e_l} or 0 based on its intersection-over-union (IoU) with the ground truth bounding box of the query object e_l belonging to class c_{e_l}. It is assigned the class c_{e_l} when its IoU ≥ 0.5 and 0 otherwise. Once the labels are assigned to the proposal boxes, the score of each region proposal with respect to the queries are generated as follows: $S_{kl} = \Theta(\widehat{p}_k^{\phi}, \widehat{e}_l^{\phi'})$, where Θ is cosine similarity and S_{kl} is the similarity score between representations of the proposal p_k and query node e_l. For each node e_l in the query graph and a set of region proposal $\mathcal{R}_u$ for image I_u, the loss function is defined as follows:

$$\begin{aligned} L(Q_u, e_l) = \sum_k \Big\{ &- \left(\mathbb{1}_{[y_k = c_{e_l}]} \ln(S_{kl})\right) \\ &- \left(\mathbb{1}_{[y_k \neq c_{e_l}]} \ln(1 - S_{kl})\right) + L_{MR}^k \Big\}. \end{aligned} \tag{9}$$

Here, L_{MR}^k is a margin loss and is defined as follows:

$$\begin{aligned} L_{MR}^k = \sum_{j=k+1} \Big\{ &\mathbb{1}_{[y_k = y_j]} max(|S_{kl} - S_{jl}| - m^- \\ &, 0) + \mathbb{1}_{[y_k \neq y_j]} max(m^+ - |S_{kl} - S_{jl}|, 0) \Big\}, \end{aligned} \tag{10}$$

where m^+ and m^- are the positive and negative margins, respectively, and y_k is the class label for the proposal p_k. The margin loss in Equation (10) takes a pair of proposals and ensures that the proposal pair that are assigned the same label has prediction probabilities closer to each other and at the same time makes the proposals with different labels wider in terms of prediction probabilities.

To select a desirable set of edges during proposal graph generation, a loss function is also defined on edges that can be constructed from the set of region proposals. For a set of N region proposals, $\binom{N}{2}$ edges connecting a pair of region

	COCO-stuff		VG-FO		SG	
	R@1	R@5	R@1	R@5	R@1	R@5
Edges removed						
node-only (Detection)	21.0	47.9	30.1	62.8	23.4	-
node-only (Localization)	33.9	57.2	29.9	53.5	34.7	62.5
Flattened triplets						
MDETR [17]	30.1	47.9	25.4	44.8	15.9	29.9
Structured Graph Query						
CRF-Based* [16]	-	-	-	-	23.9	-
Ours (VL-MPAG Net)						
1-layer	35.5	57.9	32.7	61.6	35.9	64.2
2-layers	**36.3**	**58.4**	**36.0**	**63.3**	**36.9**	**65.6**

Table 2: **Results for scene graph grounding task on COCO-stuff val and VG-FO for completely overlapping train-test categories setting.** *Due to the unavailability of the implementation of [16] at the time of submission of this paper, we only compare with reported results in their paper.

proposals can be defined. Let $\mathcal{E}$ be the set of all such edges. Consider a directed edge $h_{kj} \in \mathcal{E}$ (where k and j are source and target nodes respectively.) and its label variable z_{kj}. The variable z_{kj} is assigned a label c_{r_i} if the pair of proposals corresponding to the edge h_{kj} follows the relation r_i in the query graph. The visual representation of the said edge is subsequently scored against the relationship embedding r_i as follows: $ES_{kj,i} = \Theta(h^{\phi}_{kj}, r^{\phi'}_{i})$, where, Θ is cosine similarity, and $ES_{kj,i}$ is the score between the edge h_{kj} and the relation r_i. For the relation r_i and the set of edges $\mathcal{E}$, the loss is defined as follows:

$$L(\mathcal{E}, r_i) = \sum_{l} \Big\{ -\Big(\mathbb{1}_{[z_{kj}=c_{r_i}]} \ln(ES_{kj,i})\Big) - \Big(\mathbb{1}_{[z_{kj}\neq c_{r_i}]} \ln(1 - ES_{kj,i})\Big) \Big\}, \tag{11}$$

where, c_{r_i} is the label of relation r_i. One example of such a label is '*wearing*'. Generally, for a relation, the number of positive and negative edges have a huge imbalance (usually much more negative than positive edges), leading to poor training. To mitigate this problem, we present the following strategy to sample a more balanced set of edges.

Consider an edge (l, m) (relation) in a query scene graph. All N region proposals are scored, as defined previously, with both the nodes (e_l and e_m) present in the edge. Suppose P_l and P_m are the list of proposal sorted in decreasing order of scores with respect to e_l and e_m. We select p proposals each from P_l and P_m randomly but ensure half of them come from the top 50 of each list. From these sets of p proposals, a set of edges are formed that connect proposals from selected lists. These steps are repeated for all the edges in the query scene graph to obtain a balanced subset. Then, the loss function defined in Equation (11) is computed for these balanced subsets of edges in mini-batches. In our experiments, we empirically choose $p = 48$. We also define cross-entropy loss on the labeled (foreground or background) feature vectors of the region proposals and a regression loss on the predicted bounding box locations with respect to the ground truth bounding boxes.

Edges		1	2	3	4	5	6	7	8
VG-FO	R@1	33.8	35.8	33.5	30.7	27.9	26.0	24.6	23.2
	R@5	62.7	64.9	61.2	58.2	54.2	53.2	49.5	48.6
	#Samples	21,807	7,543	3,826	2,347	1,547	936	693	434
VG-PO-Unseen	R@1	23.9	30.0	33.7	35.1	34.7	35.7	40.1	35.7
	R@5	51.2	58.1	61.5	59.8	56.1	56.4	59.2	56.9
	#Samples	25,913	7,665	3,005	1,391	770	390	218	130

Table 3: **Effect of the size of the query on the performance of the model.** Unseen refers to the set of categories in VG-PO not used during training. (Refer to Section 4.2).

Inference: During inference, after obtaining object proposals on the image, we update query objects and proposals embeddings via message passing on the constructed graph. Different from the training, we then score the region proposals against the query objects and choose the highest-scoring proposals for each query object as the localization output.

4. Experiments and Results

4.1. Datasets, Evaluation Protocols and Baselines

We use four public datasets, namely Visual Genome [20], VRD [25], COCO-stuff [2] and SG [16] for our experiments. Among these, motivated by VRR-vg [23] and VG-150 [37], to minimize the bias due to long-tail distribution and visually-irrelevant relationship (such as a *field for plane* or *sign that says pumpkin*), we use a subset of the visual genome containing 93K image-scene graph pair for training and 40K image-scene graph pair for testing. The scene graphs in this dataset are constructed using 150 object categories and 40 predicates. We refer to this split of the visual genome as *Visual Genome-Fully Observed (or VG-FO)*. To facilitate the study of grounding unseen objects, we create a split called *Visual Genome-Partially Observed (or VG-PO)* which contains scene graphs constructed from subsets of 125 object categories during training, whereas the testing scene graphs contain subsets of additional 25 object categories. The other three datasets, i.e., VRD [25], COCO-stuff [2], and SG [16] contain (45K, 100, 70), (77K, 183, 6), and (5K, 166, 68) number of images-scene graph pairs, object categories, predicates in all. The scene graphs for COCO-stuff are constructed using protocols from [15]. We use Recall at 1 and 5 (denoted as R@1 and R@5 from here onwards) to evaluate scene graph grounding by considering an object localization as correct when its intersection over union with the ground truth bounding box is ≥ 0.5

Baselines: As there is no existing method that addresses the task of visual grounding when scene graph is used as query except a CRF-based approach [16]. Therefore, along with comparing against them, we present baselines to understand: (i) **Importance of visual relations (i.e., edges in the query) in localizing objects.** To this end,

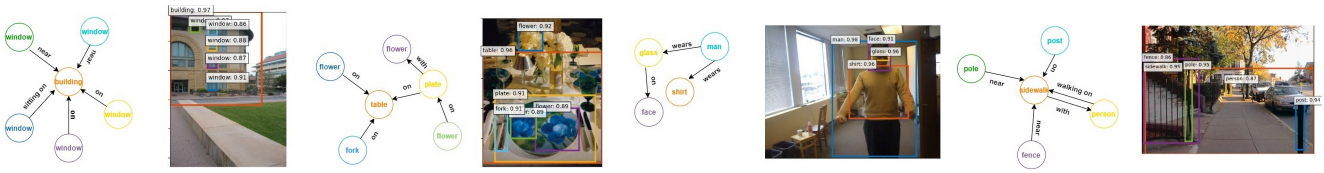

Figure 3: A selection of results on Visual Genome. Scene graph query along with the grounding results are shown side by side using the same color border for object nodes in the query graph and corresponding grounded object bounding boxes.

we present the following two baselines for edges-removed queries: (a) *Node only (Detection-based):* We detect the set of object categories in the query scene graph using Faster-RCNN [29]. Note that this model is limited to object categories seen during training. (b) *Node only (Localization-based):* In this, we obtain region proposals using faster-RCNN and then score them against the Glove word representation of each object present in the query graph to generate localizations. (ii) **Importance of structured property of the query graph.** To this end, we use flattened triplets (subject-predicate-object) obtained from the scene graph as a query in MDETR [17] – a transformer-based model for visual grounding task. To make a fair comparison with our model, we train this model only on our datasets without any pretraining and use Resnet50 as the backbone. Further, when the scene graph contains only two nodes, the problem of scene graph grounding reduces to referring relationships [19]. Thus, for such cases, we compare with state-of-the-art referring relationship methods [19, 25, 12].

4.2. Results and Discussion

We first show the results on VG-FO, COCO-stuff, and SG datasets in Table 2. We observe that VL-MPAG Net outperforms all the baselines on all datasets. The node-only baselines do not leverage the visual relationship in the scene graph query and perform poorly. The flattened scene-graph grounding approach (MDETR) does not encode the structural information in a scene graph and falls short in performance. Also, it needs to deal with language understanding challenges such as co-references, noun phrase and relationship extraction, and long-range dependency of the concepts. MDETR requires a lot of training data; therefore, to evaluate the model on the SG dataset (which contains only 4K training samples), we utilize the MDETR model trained on the VG-FO dataset. The VL-MPAG Net outperforms the CRF-based approach [16], indicating better representation learning by GNNs than CRF for the scene-graph localization task. Further, COCO-stuff has fine-grained object categories that are semantically very close (e.g., wall-wood vs. *wall-stone*). This causes inferior performance of *node only (detection)* baselines on COCO-stuff.

Effect of number of edges in the query graph: We perform grounding scene graphs experiments with varying sizes of the query scene graph. The largest scene graphs we ground on the image in our experiments contain eight edges. To analyze the performance of VL-MPAG Net with respect to scene graph size, we compute R@1 and R@5 with varying numbers of edges in the query scene graph on two splits of VG datasets used in this paper. As shown in Table 3, our method successfully grounds scene graphs even when the graph contains as large as eight edges. In the case of VG-FO, as the dataset contains fewer samples of large-size scene graphs, there is a drop in recall when the graph size becomes larger. In contrast, on grounding unseen objects (also refer to Grounding Unseen Objects towards the end of this section), a larger scene graph size helps. This result is intuitive as a large graph gives better global context, subsequently enabling the grounding of unseen objects as well.

Model	Subject		Object	
	R@1	R@5	R@1	R@5
SSAS [19]	21.5	-	24.2	-
VRD-LP [25]	31.5	38.8	34.9	40.3
CPARR [12]	49.8	69.4	**52.4**	70.2
Ours (VL-MPAG Net)	**51.6**	**79.3**	51.7	**76.1**

Table 4: **Comparison of VL-MPAG Net against the referring relations baselines** for the scenario when graph contains only two nodes on VRD dataset.

For scene graphs containing only one edge, the problem of scene graph grounding reduces to referring relationship. We directly compare our approach with state-of-the-art referring relationships methods in Table 4 on VRD dataset. Here, CPARR [12] utilizes the relation between the query nodes by combining the node prediction score with the relation prediction score at the last stage. VL-MPAG Net, instead, utilizes the relation information during the initial stages of modeling and, thereby, achieves competitive if not better R@1 and significantly better R@5 (nearly 10% and 6% better compared to the most competitive method).

Qualitative Analysis: A selection of scene graphs grounding on the VG-FO dataset is shown in Figure 3. From detailed analysis (refer to the supplementary material), we observe that VL-MPAG Net is able to localize the correct objects in a dense image containing instances of many object categories. As an example in the Figure 3, in the second example, the model is able to localize the 'flowers' printed on the 'plate' as specified in the query.

Model	Seen Categories		Unseen Categories	
	R@1	R@5	R@1	R@5
Node only (Localization)	33.2	56.6	19.6	43.1
MDETR [17]	26.2	47.1	26.4	45.7
Ours(VL-MPAG Net)				
1-layer	38.0	64.9	27.5	54.5
2-layers	**39.9**	**66.9**	**29.0**	**53.6**

Table 5: The proposed model outperforms the baselines for 'seen' and 'unseen' object categories on VG-PO dataset.

Multi-instance Localization: The representations learned for the nodes of the same class in our framework differ due to the difference in the one- or two-hop neighboring objects and relations. Therefore, our framework naturally enables multi-instance localization. Even if one- or two-hop objects and relations are the same (For example, the leftmost localization in Figure 3), our model allows localizing all the objects corresponding to each node and thereby enables multi-instance localization. However, in such rare cases, it becomes infeasible to disambiguate different instances.

Grounding Unseen Objects: Although the primary goal of this work is to address the task of grounding scene graphs on natural images, localization of unseen object categories is an auxiliary but challenging setting that we also evaluated on the VG-PO dataset in Table 5. Compared to 'seen', 'unseen' object categories suffer in performance. However, the proposed model can better capture the context between objects, leading to significantly better performance than the baselines. The *node only (loc.)* suffers the most because it does not utilize the relations that might help the localization of 'unseen' object categories. MDETR, on the other hand, uses a transformer-based model to learn the contextual embeddings and captures the context for 'unseen' categories, which in turn aid in their localization. *Node only (det.)* requires all the object categories to be known during training; hence, is dropped for comparison.

Robustness to Sparse and Incomplete Query: The scene graph grounding method must be robust against sparse and incomplete queries, not just clean ones. To demonstrate the robustness of the proposed model, we first perturb the scene-graph queries by introducing different degrees of noise and then evaluate our model against such perturbed queries. For each edge in the scene graph, we perturb the graph with a probability p by replacing the subject, object, or relation with synonyms obtained using the WordNet synsets or removing the relation from the query graph. We utilized the VL-MPAG Net model trained on the VG-FO dataset to perform this analysis. For 10% to 40% noise, we obtain the $R@1 = [30.6, 29.9, 29.0, 28.2]$ respectively, demonstrating the robustness of our model to incomplete and sparse scene graph queries.

Analysis of the message-passing steps: The message passing, in our framework, is performed on the auxiliary edges from the nodes of the query graph to the nodes of the proposal graph (AE-MP), the query graph (QG-MP), and the proposal graph (PG-MP). To better understand the effect of these message-passing steps and their order, we performed an ablation experiment by removing them one by one and changing the order of AE-MP and QG-MP on the VG-FO dataset. Results of this ablation are reported in Table 6. The first step, where message passing is performed on the auxiliary edges from the query graph to the proposal graph (AE-MP), is essential, as evident by the decrease in performance (row 2 in Table 6) when it is excluded. Further, the message passing on the query graph (QG-MP) and the proposal graph (PG-MP) helps to incorporate the structural information while learning the representation of the nodes and the edges, leading to improvement in localization performance. The order of message passing is also crucial. Performing message passing on auxiliary nodes first, followed by query graph and proposal graph respectively (AE-MP $\rightarrow$ QG-MP $\rightarrow$ PG-MP) helps VL-MPAG Net learn better conditional node representations for the proposal graph, thereby helping to achieve better localization. This is also evident in results where we observe that the above order gives a superior performance compared to one where message passing on query graph (QG-MP) is performed before message passing on auxiliary nodes (AE-MP).

AE-MP	QG-MP	PG-MP	Order	R@1	R@5
				29.9	53.5
	✓	✓		28.5	53.0
✓		✓		31.8	59.6
✓				29.9	53.7
✓	✓	✓	QG-MP $\rightarrow$ AE-MP $\rightarrow$ PG-MP	31.3	59.3
✓	✓	✓	AE-MP $\rightarrow$ QG-MP $\rightarrow$ PG-MP	**32.7**	**61.6**

Table 6: **Analysis of the message passing steps performed on the VG-FO dataset.** Here AE-MP, QG-MP, and PG-MP denote Message Passing on auxiliary visio-lingual edges, the query graph, and the proposal graph, respectively. $A \rightarrow B$ indicates that A is performed before B. The last row represents our full model.

5. Conclusions

We thoroughly studied the problem of grounding scene graphs on natural images, presented an end-to-end visio-lingual message passing-based graph neural network framework, and performed experiments on large-scale image datasets. The performance improvement over baselines confirms the efficacy of the proposed VL-MPAG Net and exhibits that better modeling of the context present in scene graphs leads to better grounding. We believe this work will revive research interests and future contributions toward the under-explored scene graph-based grounding problem.

Acknowledgments: This work is partially supported by Startup Research Grant (SRG) by the SERB, Govt. of India (File number: SRG/2021/001948) to Anand Mishra.

References

[1] Irving Biederman, Robert J. Mezzanotte, and Jan C. Rabinowitz. Scene perception detecting and judging objects undergoing relational violations. *Cognitive Psychology*, 14:143–177, 1982.

[2] Holger Caesar, Jasper R. R. Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1209–1218, 2018.

[3] Khyathi Raghavi Chandu, Yonatan Bisk, and Alan W. Black. Grounding 'grounding' in NLP. In *Findings of ACL/IJCNLP*, 2021.

[4] Ming-Ming Cheng, Shuai Zheng, Wen-Yan Lin, Vibhav Vineet, Paul Sturgess, Nigel T. Crook, Niloy J. Mitra, and Philip H. S. Torr. Imagespirit: Verbal guided image parsing. *ACM Trans. Graph.*, 34(1):3:1–3:11, 2014.

[5] Vinay Damodaran, Sharanya Chakravarthy, Akshay Kumar, Anjana Umapathy, Teruko Mitamura, Yuta Nakashima, Noa García, and Chenhui Chu. Understanding the role of scene graphs in visual question answering. *ArXiv*, abs/2101.05479, 2021.

[6] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 3837–3845, 2016.

[7] Helisa Dhamo, Azade Farshad, Iro Laina, Nassir Navab, Gregory Hager, Federico Tombari, and C. Rupprecht. Semantic image manipulation using scene graphs. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5212–5221, 2020.

[8] Martin Engilberge, Louis Chevallier, Patrick Pérez, and Matthieu Cord. Finding beans in burgers: Deep semantic-visual embedding with localization. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3984–3993, 2018.

[9] Shalini Ghosh, Giedrius Burachas, Arijit Ray, and Avi Ziskind. Generating natural language explanations for visual question answering using scene graphs and visual attention. *ArXiv*, abs/1902.05715, 2019.

[10] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1263–1272. PMLR, 2017.

[11] Francesco Giuliari, Geri Skenderi, Marco Cristani, Yiming Wang, and Alessio Del Bue. Spatial commonsense graph for object localisation in partial scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19518–19527, June 2022.

[12] Chuanzi He, Haidong Zhu, Jiyang Gao, Kan Chen, and R. Nevatia. Cparr: Category-based proposal analysis for referring relationships. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 4074–4083, 2020.

[13] Ting-I Hsieh, Yi-Chen Lo, Hwann-Tzong Chen, and Tyng-Luh Liu. One-shot object detection with co-attention and co-excitation. In *NeurIPS*, 2019.

[14] Yue Hu, Siheng Chen, Xu Chen, Ya Zhang, and Xiao Gu. Neural message passing for visual relationship detection. *CoRR*, abs/2208.04165, 2022.

[15] Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1219–1228, 2018.

[16] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3668–3678, 2015.

[17] Aishwarya Kamath, Mannat Singh, Yann LeCun, Ishan Misra, Gabriel Synnaeve, and Nicolas Carion. Mdetr - modulated detection for end-to-end multi-modal understanding. *ArXiv*, abs/2104.12763, 2021.

[18] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[19] Ranjay Krishna, Ines Chami, Michael S. Bernstein, and Li Fei-Fei. Referring relationships. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6867–6876, 2018.

[20] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123:32–73, 2016.

[21] Hsin-Ying Lee, Weilong Yang, Lu Jiang, Madison Le, Irfan Essa, Haifeng Gong, and Ming-Hsuan Yang. Neural design network: Graphic layout generation with constraints. In *ECCV*, 2020.

[22] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *ArXiv*, abs/1908.03557, 2019.

[23] Yuanzhi Liang, Yalong Bai, Wei Zhang, Xueming Qian, Li Zhu, and Tao Mei. Vrr-vg: Refocusing visually-relevant relationships. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10402–10411, 2019.

[24] Yongfei Liu, Bo Wan, Xiaodan Zhu, and Xuming He. Learning cross-modal context graph for visual grounding. *ArXiv*, abs/1911.09042, 2020.

[25] Cewu Lu, Ranjay Krishna, Michael S. Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *ECCV*, 2016.

[26] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*, 2019.

[27] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. In *EMNLP*, 2014.

[28] Zachary Ravichandran, Lisa Peng, Nathan Hughes, J. Daniel Griffith, and Luca Carlone. Hierarchical representations and explicit memory: Learning effective navigation policies on 3d scene graphs using graph neural networks. *ArXiv*, abs/2108.01176, 2021.

[29] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1137–1149, 2015.

[30] Sebastian Schuster, Ranjay Krishna, Angel X. Chang, Li Fei-Fei, and Christopher D. Manning. Generating semantically precise scene graphs from textual descriptions for improved image retrieval. In *Proceedings of the Fourth Workshop on Vision and Language, VL@EMNLP*, 2015.

[31] Sitong Su, Lianli Gao, Junchen Zhu, Jie Shao, and Jingkuan Song. Fully functional image manipulation using scene graphs in a bounding-box free way. *Proceedings of the 29th ACM International Conference on Multimedia*, 2021.

[32] Aditay Tripathi, Rajath R. Dani, Anand Mishra, and Anirban Chakraborty. Sketch-guided object localization in natural images. In *ECCV*, 2020.

[33] Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[34] Hongwei Wang, Hongyu Ren, and Jure Leskovec. Relational message passing for knowledge graph completion. In Feida Zhu, Beng Chin Ooi, and Chunyan Miao, editors, *KDD '21: The 27th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, Singapore, August 14-18, 2021*, pages 1697–1707. ACM, 2021.

[35] Sijin Wang, Ruiping Wang, Ziwei Yao, S. Shan, and Xilin Chen. Cross-modal scene graph matching for relationship-aware image-text retrieval. *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1497–1506, 2020.

[36] Yu-Siang Wang, Chenxi Liu, Xiaohui Zeng, and Alan Loddon Yuille. Scene graph parsing as dependency parsing. In *NAACL*, 2018.

[37] Danfei Xu, Yuke Zhu, Christopher Bongsoo Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3097–3106, 2017.

[38] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph R-CNN for scene graph generation. In Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, editors, *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part I*, volume 11205 of *Lecture Notes in Computer Science*, pages 690–706. Springer, 2018.

[39] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. *ArXiv*, abs/1808.00191, 2018.

[40] Wei Yang, Xiaolong Wang, Ali Farhadi, Abhinav Gupta, and Roozbeh Mottaghi. Visual semantic navigation using scene priors. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[41] Zhou Yu, Jun Yu, Chenchao Xiang, Zhou Zhao, Qi Tian, and Dacheng Tao. Rethinking diversified and discriminative proposal generation for visual grounding. *ArXiv*, abs/1805.03508, 2018.

[42] Li Zhang, Dan Xu, Anurag Arnab, and Philip H. S. Torr. Dynamic graph message passing networks. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 3723–3732. Computer Vision Foundation / IEEE, 2020.

[43] Zhixuan Zhang, Chi Zhang, Zhenning Niu, Le Wang, and Yuehu Liu. Geneannotator: A semi-automatic annotation tool for visual scene graph. *ArXiv*, abs/2109.02226, 2021.

[44] Bo Zhao, Lili Meng, Weidong Yin, and Leonid Sigal. Image generation from layout. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8576–8585, 2019.

[45] Bohan Zhuang, Qi Wu, Chunhua Shen, Ian D. Reid, and Anton van den Hengel. Parallel attention: A unified framework for visual object discovery through dialogs and queries. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4252–4261, 2018.