This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Exploring Context and Visual Pattern of Relationship for Scene Graph Generation

Wenbin Wang[1,2], Ruiping Wang[1,2], Shiguang Shan[1,2,3], Xilin Chen[1,2]
[1]Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences (CAS),
Institute of Computing Technology, CAS, Beijing, 100190, China
[2]University of Chinese Academy of Sciences, Beijing, 100049, China
[3]Peng Cheng Laboratory, Shenzhen, 518055, China
wenbin.wang@vipl.ict.ac.cn, {wangruiping, sgshan, xlchen}@ict.ac.cn

Abstract

*Relationship is the core of scene graph, but its prediction is far from satisfying because of its complex visual diversity. To alleviate this problem, we treat relationship as an abstract object, exploring not only significative visual pattern but contextual information for it, which are two key aspects when considering object recognition. Our observation on current datasets reveals that there exists intimate association among relationships. Therefore, inspired by the successful application of context to object-oriented tasks, we especially construct context for relationships where all of them are gathered so that the recognition could benefit from their association. Moreover, accurate recognition needs discriminative visual pattern for object, and so does relationship. In order to discover effective pattern for relationship, traditional relationship feature extraction methods such as using union region or combination of subject-object feature pairs are replaced with our proposed intersection region which focuses on more essential parts. Therefore, we present our so-called Relationship **C**ontext - **I**nter**S**e**C**tion Region (**CISC**) method. Experiments for scene graph generation on Visual Genome dataset and visual relationship prediction on VRD dataset indicate that both the relationship context and intersection region improve performances and realize anticipated functions.*

1. Introduction

Scene graph helps higher level scene understanding. Recently, a number of works [41, 20, 48, 42, 19, 27, 7, 18, 49, 21, 23, 51, 43], have focused on discovering relationships between objects or generating a graph representation for a scene, which contains objects as nodes and their relationships as edges. Besides, scene graph has evolved as a promising alternative for high-level intelligence vision tasks, such as image captioning [24, 40, 45], image generation [13], and visual question answering [2, 38, 39, 40]. However, scene graph generation still remains a challenging problem due to complexity of predicting pair-wise relationships even if object categories and locations are all given. Although previous works have proposed a series of techniques to improve relationship prediction, visual pattern and contextual information, two key aspects in object recognition, are still not considered profoundly for relationship.

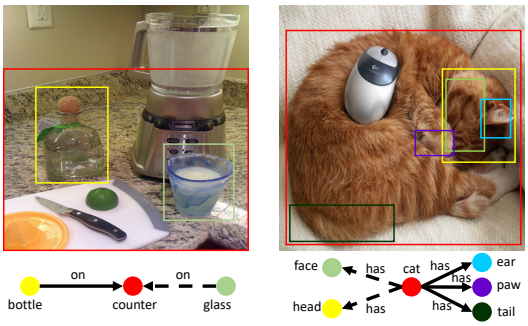


Figure 1. Examples of scene graph. All the scene graphs are generated from our baseline method [41] given ground truth objects. Dotted arrow means that the model misses this relationship, while solid one is detected correctly.

Let's firstly pay attention to contextual information, which is never exploited for relationship. Why should we consider it? In Fig.1, the relationships between *glass* and *counter*, *cat* and *face*, as well as *cat* and *head* are missed. In fact, a number of same relationships (*e.g. bottle on counter*, *cat has ear*) nearby have been detected correctly. In other words, while predicting a specific relationship, current methods only focus on the pair of regions with which it relates, but ignore other relationships which may be helpful for reasoning itself. Once the corresponding object pairs could not provide strong enough evidence for relationship inference, the methods would fail.

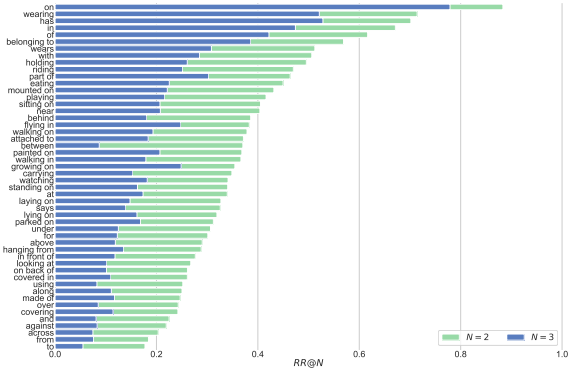


Figure 2. The fraction of images in which a relationship appears no less than N times, denoted by $RR@N$. Green bars are for $N = 2$ while blue bars are for $N = 3$.

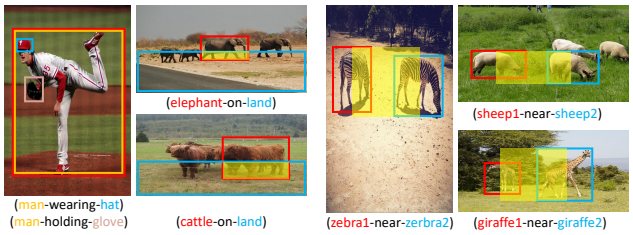


Figure 3. (a) The left image shows the high probability of overlap of union regions. The union regions of *man* (yellow box) and *hat* (blue box), *man* and *glove* (pink box) are the same (red box). Images on right show a case of intersection region (yellow mask) when two boxes are intersectant. (b) A case of intersection region when two boxes are disjoint.

To further expose the underlying occurrence pattern of relationship, we examine the fraction of images in which a relationship appears no less than N times, denoted by "Repetition Rate ($RR@N$)", using Visual Genome dataset [15] (VG) which contains more than 40k annotated unique relationships for over 100k images. As long tail distribution exists and most infrequent relationships hardly appear in normal scenes, we investigate 50 most common relationships. As shown in Fig.2, repetition is a ubiquitous phenomenon for relationships in manual annotations. What leads to this fact? As we know, the number of object categories are much more than that of relationship categories. Different object pairs could be described with the same relationship as long as they share similar visual patterns. On the other hand, there exist lots of fixed phrase structures such as "object1-has-object2" (*e.g.* "elephant-has-head", and "elephant-has-ear"), which is also indicated in [48]. As a result, many relationships tend to repeat in images. These observations are consistent with humans language habbits.

From the above observations, there exists strong association among relationships, which encourages us to make use of context to capture it. Context has been widely utilized for object-oriented tasks in the form of comprehensible object co-occurrence [4, 5, 48, 22] and the approaches of modeling object context varies. Different from objects, direct and explicit association between two relationships is not easy to model, thus we hope to gather all the relationships information to make themselves establish implicit connection. By this way, semantics and visual patterns of relationships could be reasoned and improved respectively under the guidance of mutual influences. Inspired by [4], we use memory to construct such context for relationships, where all relationships information is stored and reasoning process happens. We will show that the relationship context indeed functions and captures the frequent repetition law.

Apart from contextual information, visual pattern is another key aspect for object, and so does relationship. To the best of our knowledge, all of current works obtain relationship features either from union region [41, 18], which is the minimal closure of subject and object region, or combination of subject and object features [49]. Such combination-like representations may not expose the real visual pattern of relationship and mainly have two drawbacks. Firstly, a large number of union regions overlap with each other [19]. The left image in Fig.3(a) gives an example. The relationship features are too similar for the models to distinguish. On the other hand, the subject and object areas contain too much object information. As a result, the models may infer the relationship mainly depending on the objects instead of relationship pattern itself [43]. However, relationships especially geometric predicates (*e.g.* *on*, *in*) are almost not dependent on object categories. In this work, we hope to separate visual pattern of relationship from object as much as possible. We propose a simple but effective region, intersection region, for relationship feature extraction. As shown in the right two images in Fig.3(a) (two boxes are intersectant) and three images in Fig.3(b) (two boxes are disjoint), the interactive parts of subjects and objects are more likely to reveal the visual pattern of relationship because although the objects vary, the visual patterns in these regions are similar. Experiments on VG and VRD dataset [23] demonstrate the effectiveness of our method.

2. Related Works

Scene Graph Generation. Scene graph is firstly mentioned in [14] for image retrieval. Recently, a number of approaches [23, 7, 18, 49, 50, 21, 51, 27, 47, 31, 41, 20, 48, 42, 19, 43] are proposed to detect objects and predict relationships concurrently. Most of them shed light on message passing [41] between two related objects or the object and its corresponding relationships. The effectiveness of this message passing mechanism as well as its variants [42, 19, 20, 18, 48] is proven. In our work, we especially focus on message passing between relationships, creatively exploiting implicit association among relationships which is helpful for prediction.

Context Modeling. Context modeling and reasoning [16, 28, 30, 32, 8, 44, 34, 3, 12, 37] is one of the most helpful approaches for scene or object recognition. A variety of previous works on scene understanding [36, 17], object recognition [4, 5, 22, 11], attributes reasoning [9, 29], human-object interaction [44], action recognition [25], have benefited from context. However, context is seldom considered in scene graph generation or visual relationship detection tasks. Zellers *et al*. [48] makes an early attempt to use object context for scene graph generation. While our proposed relationship context, is totally different from object context used in [48]. We take a further step to demonstrate that relationship context is as nonnegligible as object context and even plays a more significant role in relationship-centric tasks compared with object context.

Relationship Feature Extraction. Almost all published scene graph generation or visual realtionship detection methods have to construct initial feature for relationship. The most general approaches include computing the union region [41, 18, 7] and feeding it to a local feature extraction module (e.g. RoI pooling layer [10]), or combining the subject and object features [49]. These methods are intuitive and work but either lack discrimination or rely heavily on object information. Our proposed intersection region concentrates on more essential part and is closer to the real visual pattern of relationship.

3. Approach

Our goal is not only to especically construct relationship context apart from object context to capture the hidden association among relationships, but also to discover more discriminative visual pattern for them. To this end, our method, *Relationship **C**ontext - **I**nter**S**e**C**tion Region (**CISC**)*, is devised which will be described in following subsections.

3.1. Basic Scene Graph Model

Our framework is based on a basic scene graph model which refines representations for objects and relationships with explicit message passing mechanism. Therefore, we start by describing a general skeleton for it.[1]

In a basic scene graph model, objects and relationships are modelled separately for $|C|$ object classes and $|R|$ relationship classes. They can be regarded as nodes v in a virtual graph $G = (V = V^O \bigcup V^R, E)$, as shown in the inner dotted box in Fig.4(a), where $v^O \in V^O$ denotes object, $v^R \in V^R$ denotes relationship, and edge $e = (v_i^O, v_j^O) \bigcup (v_i^O, v_{ij}^R) \bigcup (v_j^O, v_{ij}^R) \in E$ means that if object i and j are related, there are edges between v_i^O and v_j^O, v_i^O and v_{ij}^R, as well as v_j^O and v_{ij}^R. Each node has its own feature $\mathbf{f}$ and broadcasts its message to neighbors to instruct them to refine features. The brown dotted bidirectional arrows in Fig.4(a) demonstrate the message passing process.

In practice, supposing $\mathbf{f}_i^O \in \mathbb{R}^D$ and $\mathbf{f}_j^O \in \mathbb{R}^D$ are features[2] of two object candidates (obtained from region proposal methods, *e.g*. RPN [33]) associated with v_i^O and v_j^O, and $\mathbf{f}_{ij}^R \in \mathbb{R}^D$ represents the relationship feature associated with v_{ij}^R, the message passing procedure can be written as:

$$\mathbf{m}_i^O = G^O\left(\sum_{j \in \mathcal{N}_i^O} M^{O\to O}(\mathbf{f}_j^O),\ \sum_{j \in \mathcal{N}_i^O} M^{R\to O}(\mathbf{f}_{ij}^R)\right), \quad (1)$$

$$\mathbf{m}_{ij}^R = G^R\left(M^{O\to R}(\mathbf{f}_i^O),\ M^{O\to R}(\mathbf{f}_j^O)\right), \quad (2)$$

$$\mathbf{f}_i^O \leftarrow U^O(\mathbf{f}_i^O,\ \mathbf{m}_i^O), \quad (3)$$

$$\mathbf{f}_{ij}^R \leftarrow U^R(\mathbf{f}_{ij}^R,\ \mathbf{m}_{ij}^R), \quad (4)$$

where $\mathbf{m}_i^O \in \mathbb{R}^D$ and $\mathbf{m}_{ij}^R \in \mathbb{R}^D$ denote messages received by node v_i^O and v_{ij}^R respectively. $\mathcal{N}_i^O$ stands for neighbors of v_i^O. $M^{O\to O}$, $M^{R\to O}$, and $M^{O\to R}$ are message processing functions that extract useful information from node features. Their superscripts indicate the direction of message passing (*e.g*. $R \to O$ denotes "from relationship to object"). G^O and G^R represent gathering functions which integrate messages from sources. U^O and U^R are update functions for object and relationship respectively. After message passing process, the refined features could be used to make predictions. In next subsection, we will build context based on this universal skeleton.

3.2. Relationship Context Construction

With representations of objects and relationships obtained from basic scene graph model, context is able to be constructed. However, different from objects which have co-occurrence association, it is difficult to model explicit and interpretable association between any two relationships. Therefore, the most feasible way is to construct relationship context implicitly, which can be considered as a remedy of missing of message passing between relationships in basic scene graph model. On the other hand, we also hope the relationship context to keep the 2-dimensional spatial structure of an image so that a specific relationship can be affected by surrounding similar relationships if there exist. Memory [4, 5] meets our demand. In [4, 5], memory is used for object context construction. Information of previously detected objects is saved into the memory, which provides context for further object reasoning. Supposing there are N object instances $\mathcal{O} = [O_1, O_2, \ldots, O_N]$ to be detected given an image $\mathcal{I}$. Then an iterative detection model $\mathcal{M}$ is expected to maximize the log-likelihood:

$$\mathcal{L}_O \approx \log \mathbb{P}\left(O_{1:N}^{(t)} | \mathcal{S}_{1:N}^{(t-1)}, \mathcal{M}, \mathcal{I}\right), \quad (5)$$

[1] We do not differentiate "subject" and "object" but use "object" uniformly instead. We use "predicate" to refer to a certain relation.

[2] The feature is a 1-dimensional vector with size D when referring to message passing or prediction process, while it is a tensor with spatial size in memory updating process, except that there is extra explanation.

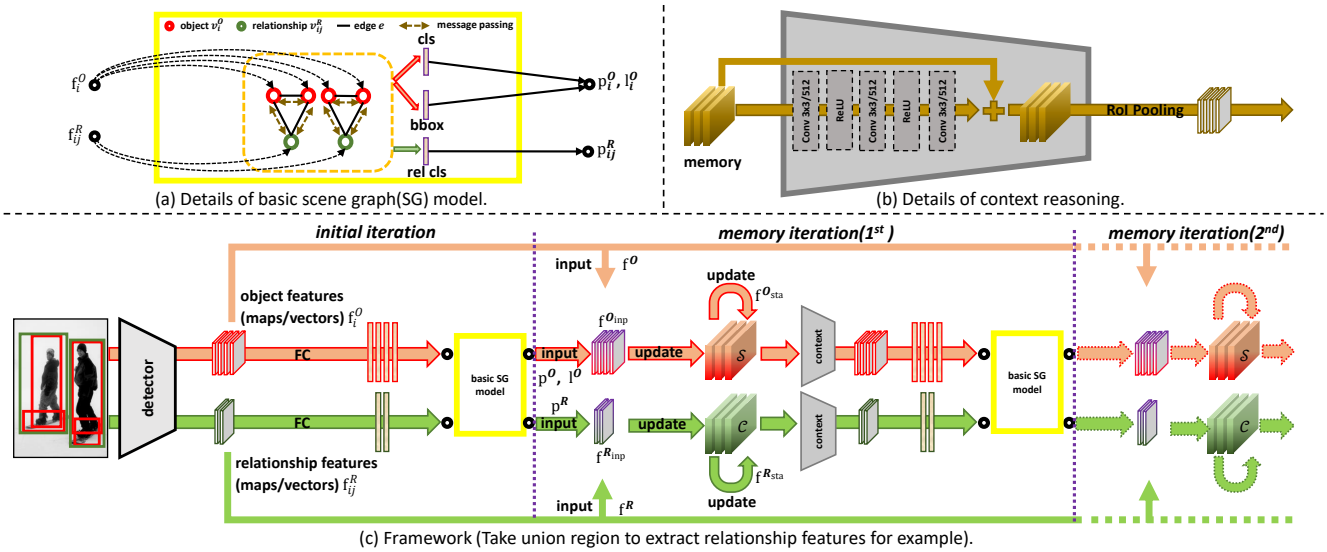


Figure 4. (a) Details of a basic scene graph model. Message passing described in Eq.(1-4) and prediction are executed through the constructed virtual graph G shown in the inner dotted box. (b) Architecture of context reasoning network. (c) The framework of our method. After obtaining initial visual features for objects and relationships, the initial iteration is triggered and produces predicted information. In each round of later memory iterations, the predicted information together with their initial visual features are used to update the memories. Then two memories take the responsibility, which will conduct context reasoning process and provide updated features for further predictions.

where $O_{1:N}^{(t)}$ stands for the prediction of all objects at timestep t and memory $\mathcal{S}_{1:N}^{(t-1)}$ encodes information of all objects at last timestep, $t-1$. $\mathcal{S}_{1:N}^{(0)}$ is an empty memory. In practice, $\mathcal{S}$ is a three-dimensional tensor with shape $h \times w \times c$. h and w are the same as the spatial size of the feature map of image $\mathcal{I}$ processed by a feature extraction network. c is depth size so that the memory stores extra useful information at each spatial location.

Naturally, we consider constructing relationship context with memory. Let $\mathcal{C}$ denote relationship memory. K relationships $\mathcal{R} = [R_1, R_2, R_3, \ldots, R_K]$ need to be classified. $\mathcal{C}_{1:K}^{(t)}$ encodes information of all relationships at timestep t and $\mathcal{C}_{1:K}^{(0)}$ is empty. We extend the detection model $\mathcal{M}$ to our whole framework. The relationship prediction part of $\mathcal{M}$ is expected to maximize:

$$\mathcal{L}_R \approx \log \mathbb{P}\left(R_{1:K}^{(t)} | \mathcal{C}_{1:K}^{(t-1)}, \mathcal{M}, \mathcal{I}\right). \quad (6)$$

Next we describe the whole framework, which is depicted in Fig.4(c). The object representations $\mathbf{f}_i^O$ obtained from the front-end object detector and features $\mathbf{f}_{ij}^R$ of related pairs of objects acquired with relationship feature extraction methods are instantly fed into the basic scene graph model and make predictions as shown in Fig.4(a). The predicted information includes the object class scores $\mathbf{p}^O \in \mathbb{R}^{N \times |C|}$ and locations $\mathbf{l}^O \in \mathbb{R}^{N \times 4|C|}$ (4 indicates four coordinates of bounding boxes), and relationship class scores $\mathbf{p}^R \in \mathbb{R}^{K \times |R|}$. This is the initial iteration.

The later memory iterations begin with memory updating. Firstly, we hope to remember as much known information as possible in memories. Therefore, the inputs to memories $\mathbf{f}^{O_{\text{inp}}}, \mathbf{f}^{R_{\text{inp}}}$ should contain fixed initial visual features together with predicted information (denoted by four "input" arrows in Fig.4(c)):

$$\mathbf{f}^{O_{\text{inp}}} = \text{ReLU}(\text{Fc}(\mathbf{p}^O) + \text{Fc}(\mathbf{l}^O) + \text{Conv}^{1\times1}(\mathbf{f}^O)), \quad (7)$$

$$\mathbf{f}^{R_{\text{inp}}} = \text{ReLU}(\text{Fc}(\mathbf{p}^R) + \text{Conv}^{1\times1}(\mathbf{f}^R)), \quad (8)$$

where the fully connected layers and convolutional layers are used to unify dimensions. Secondly, let's consider details of memory updating (denoted by four "update" arrows in Fig.4(c)). Since the memories should not forget previously obtained information, we exploit the update mechanism of GRU [6] which is a kind of RNN. Thus we regard the memories as GRU cells. In a GRU cell, previously acquired information is stored as internal state, which can also influence the output. Similarly, the states of memories, denoted by $\mathbf{f}^{O_{\text{sta}}}$ and $\mathbf{f}^{R_{\text{sta}}}$, are obtained by applying RoI pooling operation to the memories. Finally the new features are computed with GRU and memories are updated with inverse RoI pooling opearation (similar to the operation mentioned in [46], which puts the features back to their original spatial positions):

$$\mathbf{f}^{\bullet_{\text{new}}} = z * \mathbf{f}^{\bullet_{\text{sta}}} + (1-z) * \sigma(W_U \mathbf{f}^{\bullet_{\text{inp}}} + W_H(r * \mathbf{f}^{\bullet_{\text{sta}}})), \quad (9)$$

$$\mathcal{S} = \text{InvRoIP}(\mathbf{f}^{O_{\text{new}}}), \ \mathcal{C} = \text{InvRoIP}(\mathbf{f}^{R_{\text{new}}}) \quad (10)$$

where $\bullet$ stands for O or R, z and r are update and reset gate in standard GRU, W_U and W_H are learnable convolutional parameters, σ is sigmoid function and $*$ denotes element-wise product. InvRoIP denotes inverse RoI Pooling.

Now the object memory and relationship memory take the responsibility, where context reasoning process is con-

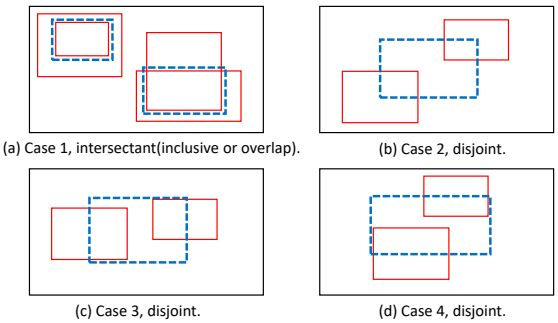


Figure 5. Four cases of intersection region. Red solid boxes are object boxes while the blue dotted boxes are our designed intersection region. Case 1 contains two situations.

ducted. As the memories contain both semantic and visual information, convolution is used to help integrate them and spread information of a certain object or relationship to surroundings. Similar to [5], context reasoning is realized with three 3×3 convolutions and residual structure, as shown in Fig.4(b). Especially for relationships, this process makes use of a large number of similar visual patterns and helps the model learn better representations. After context reasoning, object and relationship features are obtained from these two memories and used for further predictions.

3.3. Intersection Region

As introduced in Sec.1, current methods for extracting relationship features are either lacking in discrimination or seriously dependent on objects. We propose intersection region shown in Fig.5 which focuses on more essential part and reduces distractive object information. We elaborately devise it considering both the intersectant and disjoint cases.

Given bounding boxes $[x_1^i, y_1^i, x_2^i, y_2^i]$ and $[x_1^j, y_1^j, x_2^j, y_2^j]$ for two objects i and j where x_1, x_2 are horizontal boundaries and y_1, y_2 are vertical boundaries, we firstly judge their relative position. Let (c_x^i, c_y^i) and (c_x^j, c_y^j) be center points of two boxes, w^i, h^i, w^j, h^j be widths and heights. We give two auxiliary conditions for judgement:

$$|c_x^i - c_x^j| \geq \frac{w^i + w^j}{2} \tag{11}$$

$$|c_y^i - c_y^j| \geq \frac{h^i + h^j}{2} \tag{12}$$

There are four cases:

1. Intersectant. The intersection box is directly obtained:
$$\mathbf{B}_{isc} = [\max(x_1^i, x_1^j), \max(y_1^i, y_1^j), \min(x_2^i, x_2^j), \min(y_2^i, y_2^j)] \tag{13}$$
2. Disjoint, satisfies condition (11) and (12):
$$\mathbf{B}_{isc} = [\min(c_x^i, c_x^j), \min(c_y^i, c_y^j), \max(c_x^i, c_x^j), \max(c_y^i, c_y^j)] \tag{14}$$
3. Disjoint, satisfies condition (11) but violates (12):
$$\mathbf{B}_{isc} = [\min(c_x^i, c_x^j), \min(y_1^i, y_1^j), \max(c_x^i, c_x^j), \max(y_2^i, y_2^j)] \tag{15}$$
4. Disjoint, satisfies condition (12) but violates (11):
$$\mathbf{B}_{isc} = [\min(x_1^i, x_1^j), \min(c_y^i, c_y^j), \max(x_2^i, x_2^j), \max(c_y^i, c_y^j)] \tag{16}$$

In the experiment section, we will introduce how to use and evaluate our intersection region in practice.

4. Experiments

In following subsections we firstly clarify experimental settings including datasets, evaluation metrics, and implementation details. Then we show the experiment results.

4.1. Experiment Settings

Datasets. Visual Genome is the largest dataset annotated with scene graphs. However, different splits are used in previous works. We follow the split in [41] which is the most common used. The split contains 75,651 images for training and 32,422 images for testing. The most frequent 50 relationship categories and 150 object categories are selected to be the predicted targets. Besides, VRD [23] is a standard dataset for visual relationship detection, containing 4,000 images for training and 1,000 images for testing. 100 object categories and 70 relationship categories are considered.

Evaluation. We adopt three universal evaluation tasks for scene graph generation: (1) **predicate classification** (PREDCLS): given ground truth categories and locations of any two objects, predict their relationship, (2) **scene graph classification** (SGCLS): given ground truth locations of any two objects, predict their categories and relationship, and (3) **scene graph generation** (SGGEN): detect objects and predict pair-wise relationships, and objects who have at least 0.5 IoU overlap with their ground truth boxes are considered to be correctly detected. All evaluation modes use recall@K metrics, where K maybe 20, 50 or 100.

4.2. Implementation Details

Choice of Basic Scene Graph Model. In Sec.3.1 we give a general skeleton of basic scene graph model. In practice, a model can be selected as long as the message passing mechanism described by Eq.(1-4) is applicable. We choose the model proposed in [41] for its favorable performance, great popularity and easy implementation.

Models and Training Details. In the experiments, we compare the results between union region and intersection region. Besides, in order to explore for a better performance, we further try to combine these two types of features. Faster-RCNN [33] with VGG-16 [35] backbone is selected as our front-end object detector for fair comparison. After the detector is trained and its layers are frozen, the whole framework is then trained on ground truth scene graph annotations. Furthermore, we also try to assemble the predictions from each iteration with **attention** mechanism

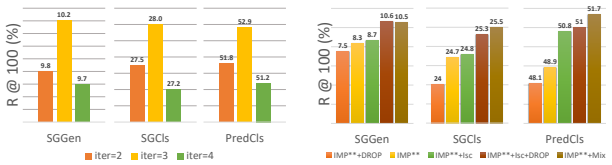
[26]. Therefore, when making a prediction in each iteration, an extra attention weight is predicted at the same time. More details can be found in supplementary materials. The source codes are implemented with Tensorflow[3] [1].

4.3. Quantitative Results

We compare the following models and present main quantitative results in Table 1. **Mem**: Our context-utilized model. It uses union region to extract relationship features. **Mem+Isc**: Our context-utilized model which replaces union region with our intersection region. **Mem+Mix**: Our proposed full model which combines two types of relationship features. **Mem+Mix+Attention**: Based on model Mem+Mix, we further assemble predictions from each iteration with predicted attention weights in order to obtain a best result. **IMP** [41]: Our *baseline* which uses union region to extract relationship features. We reimplement this model and re-train it using our object detector. In Table 1, the results of this model reported in [41] and [42] are presented together with ours. **IMP+Isc** and **IMP+Mix**: Replace the union region used in IMP with intersection region or combination version respectively. **Graph-RCNN** [42]: It is also a scene graph generation model based on message passing. **VRD** [23]: We present its scene graph generation results reported in [41]. **Pixel2Graph** [27]: We report its results according to [48]. **MSDN** [20]: The VG split it uses is different from ours. We train and evaluate it on our data split and report the original and our reimplemented results.

From Table 1, results of our reimplemented IMP model are close to or better than those of original version and reimplemented version by [42] under most metrics, which means that our reimplementation is correct and the improvements mentioned below are from our proposed method. Firstly, through the comparisons between Mem and IMP**, Mem+Isc and IMP**+Isc, Mem+Mix and IMP**+Mix, it preliminarily indicates that the usage of context is effective in helping the model recognize objects and relationships. We will further compare the importance between relationship context and object context and evaluate the function of relationship context in following subsections. On the other hand, IMP**+Isc performs better than IMP**, and Mem+Isc outperforms Mem under most metrics. It shows effectiveness of our intersection region. Finally, the assembled models, IMP**+Mix, Mem+Mix, and Mem+Mix+Attention, further boost the performance. It is noteworthy that since our basic scene graph model is IMP which limits the upper bound of performance, our models cannot surpass some methods like Graph-RCNN or Pixel2Graph under some metrics. However, results of our assembled model, Mem+Mix+Attention, are close to them and even better under some metrics.

[3]Our source codes are available at http://vipl.ict.ac.cn/resources/codes.



(a) (b)

Figure 6. (a) Results of using various iterations for model **Mem**. (b) Performances of utilizing different methods to extract relationship features. Results of three tasks are shown under R@100 metric.

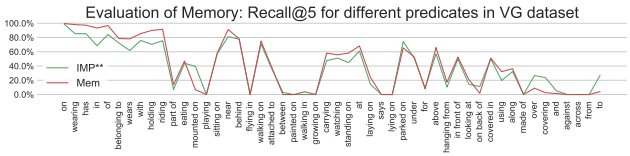


Figure 7. The per-type recall@5 of classifying individual predicate tested on VG dataset. The predicates are listed in descending order from left to right according to their repetition rates(RR@2).

4.4. Evaluation of Memory

Ablation Study. To compare the importance of object memory and relationship memory, we consider ablation experiments in Table 1. Mem\relmem and Mem\objmem stands for dropping the relationship memory module and object memory module from Mem respectively. The results suggest that the removal of relationship memory does more harm to performance than removal of object memory. It implies that the association among relationships is nonnegligible and even more important than that among objects for scene graph generation tasks.

Multiple Iterations Analysis. We investigate the performances of using various iterations for model Mem as shown in Fig.6(a). We find that 3 iterations are the best. Since the memories are empty at the first iteration, it actually only takes 2 iterations for memories to capture the context. More iterations may enhance noise.

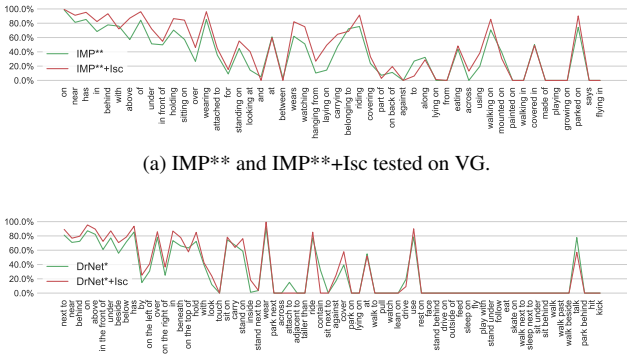
Predicate Prediction. In order to explore what context the relationship memory module actually captures, we evaluate per-type recall@5 of classifying individual predicate, following [41]. In Fig.7, the per-type recall rates for IMP** and Mem tested on VG dataset are listed in descending order from left to right according to the predicate repetition rate (RR@2). We can find that the relationship memory improves most results of predicates which have higher repetition rate (near the left side in Fig.7) despite a few outliers. For these predicates, it is easier for the memory module to extract similar patterns and learn stronger representations. While for some predicates with low repetition rate, the contribution of relationship memory is limited. And on some outliers, *e.g.* "mounted on", "parked on ", it fails mainly be-

	Model	Scene Graph Generation			Scene Graph Classification			Predicate Classification		
		R@20	R@50	R@100	R@20	R@50	R@100	R@20	R@50	R@100
models	VRD [23]	-	0.3	0.5	-	11.8	14.1	-	27.9	35.0
	IMP [41]	-	3.4	4.2	-	21.7	24.4	-	44.8	53.0
	IMP* [42]	-	6.4	8.0	-	20.6	22.4	-	40.8	45.2
	Pixel2Graph [27, 48]	6.5	8.1	8.2	18.2	21.8	22.6	47.9	54.1	55.4
	MSDN [20]	-	10.7	14.2	-	24.3	26.5	-	67.0	71.0
	MSDN* [20]	-	11.1	14.0	-	-	-	-	-	-
	Graph-RCNN [42]	-	11.4	13.7	-	29.6	31.6	-	54.2	59.1
	IMP**(baseline)	4.2	6.8	8.3	15.7	21.4	24.7	29.9	40.4	48.9
	IMP**+Isc	4.7	7.3	8.7	16.6	21.9	24.8	31.0	43.0	50.8
	IMP**+Mix	5.3	8.0	10.5	17.3	22.6	25.5	31.7	44.2	51.7
	Mem	4.8	7.6	10.2	19.5	25.0	28.0	32.3	44.9	52.9
	Mem+Isc	5.0	7.9	10.5	19.4	25.0	28.0	31.9	45.2	52.4
	Mem+Mix	6.0	9.4	11.9	19.7	25.0	27.7	33.3	45.9	53.0
	Mem+Mix+Attention	**7.7**	**11.4**	**13.9**	**23.3**	**27.8**	**29.5**	**42.1**	**53.2**	**57.9**
Ablations	Mem\relmem	4.5	7.3	9.7	19.0	24.5	27.7	31.9	44.0	51.9
	Mem\objmem	4.8	7.4	10.0	19.3	25.0	27.9	32.0	44.6	52.5
	Mem	4.8	7.6	10.2	19.5	25.0	28.0	32.3	44.9	52.9

Table 1. Results table on Visual Genome test set. All numbers are in %. IMP*: results reimplemented by [42]. IMP**: results reimplemented by us. MSDN*: The results reimplemented by us on our VG data split. Evaluation details about PREDCLS and SGCLS in MSDN are not released.

	Predicate Classification	
model	R@50	R@100
DrNet [7]	80.78	81.90
DrNet*	78.12	79.01
DrNet*+Isc	78.37	79.43
DrNet*+Mix	78.78	79.62

Table 2. Results on VRD test set. DrNet* denotes our reimplementation using union region. DrNet*+Isc and DrNet*+Mix use intersection region or mixture version.



(a) IMP** and IMP**+Isc tested on VG.

(b) DrNet* and DrNet*+Isc tested on VRD.

Figure 8. The per-type recall@5 of classifying individual predicate. The predicates are listed from left to right according to their degree of dependence to certain subject-object pairs (left side means less dependence).

cause these predicates are overshadowed seriously by other semantically similar predicates, which may be ascribed to annotation bias. Despite this, It is undeniable that the relationship memory captures the repetition law sucessfully and helps the model learn better representations on most predicates with high repetition rate and unambiguous semantics.

4.5. Evaluation of Intersection Region

Results in Table 1 have shown effectiveness of intersection region. To further validate the universality, we conduct another experiment for visual relatonship detection on VRD dataset using model in [7]. We reimplement part of this model, feeding it with ground truth objects and only predicting the relations. The original model contains several modules which are trained separately. We train it end-to-end. Results are shown in Table 2. Although the improvement is not so obvious because of lots of ambiguous predicates, it still proves the universality of intersection region.

Predicate Prediction. We explore the effect of intersection region on each predicate. We firstly compute the number of subject-object pairs that each predicate associates with. The predicate with larger number means that it can be used to describe relationships for more types of subject-object pairs, and thus has less dependence on a certain type of pair. The per-type recall@5 rates for comparisons between (IMP**, IMP**+Isc), and between (DrNet*, DrNet*+Isc) are shown in Fig.8. The predicates are listed in a descending order from left to right according to the num-

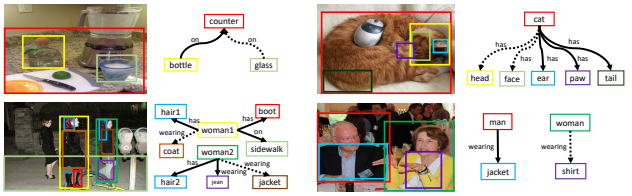


Figure 9. Examples of scene graph under the setting of PREDCLS metric. All arrows (including dotted and solid types) are ground truth relationships and detected correctly by Mem. Dotted arrows stand for missed ones of IMP**.

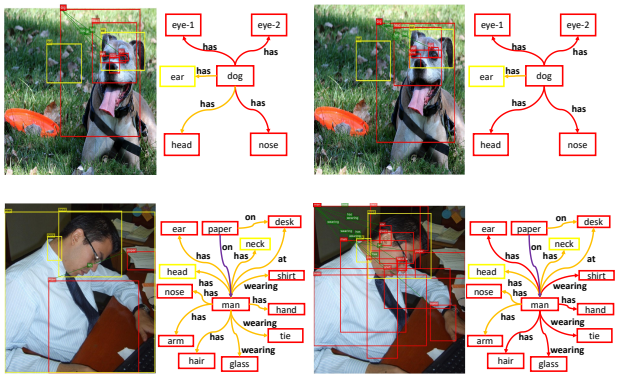


Figure 10. Scene graph generation examples under the setting of SGGEN metric for comparing IMP** with Mem. In each row, the left image and scene graph are generated by IMP** while the right ones are generated by Mem. In images and scene graphs, red boxes are predicted and overlap with the ground truth, yellow boxes are ground truth with no match. In scene graphs, red edges are true positives, orange edges are false negatives, purple boxes and edges are false positives. Some yellow boxes in scene graphs which do not exist in images mean that they are detected correctly but the model fails in detecting their relationships with any other objects.

ber mentioned above. No matter in the VG or VRD dataset, the predicates with less dependence are almost all geometric types. The intersection region especially contributes to prediction of these predicates because features from intersection region are closer to the real visual patterns of predicates and are less likely to be distracted by object information, and at the same time geometric predicates rely less on object categories compared with semantic predicates.

Feature-level Ablation Study. Since the traditional union region indeed covers our intersection region, it's natural to ask a question: is it the intersection region that plays a significant role in relationship prediction? We conduct the feature-level ablation study. Apart from the model IMP**, IMP**+Isc, and IMP**+Mix mentioned above, we further evaluate another two models. One is to drop features in intersection region from union region by setting the features in intersection region to 0 (**IMP**+DROP**) and the other one is to combine features in intersection region with DROP (**IMP**+Isc+DROP**). The results are shown in Fig.6(b). IMP**+DROP declines from IMP** while IMP**+Isc+DROP performs similarly to IMP**+Mix. It further justifies the key importance of intersection region.

4.6. Qualitative Results

Qualitative examples for comparing the IMP** and Mem under the setting of PREDCLS task are shown in Fig.9. The results show higher predicate recall rates of our method. What's more, it is obvious that although the object categories with which a relationship associates are different, the visual patterns of the relationship are similar. The relationship context gathers these similar patterns to improve relationship representations and enhance recognition capability. In Fig.10 we show some generated scene graphs using Mem and IMP** on VG test images for contrastive analysis. It shows that our method obtains higher recall with the help of context. More qualitative results can be found in supplementary materials, where we also provide examples for comparing IMP** and IMP**+Isc to show the superiority of intersection region.

5. Conclusion

In this work, we regard relationships as abstract objects in scene graph generation task, considering their visual patterns and contextual information. We discover that repetition is a ubiquitous phenomenon among relationships, hence we construct context for relationships apart from objects. Experiments show that the relationship context indeed captures the repetition law and even more helpful for generating scene graphs compared with object context. What's more, intersection region is proposed to help recognize relationships relying more on their own visual patterns instead of object information. From our evaluations, our methods are universal and have potential to be used with other better basic scene graph models. Despite our efforts on solving this task, there still exist some problems which are worthy of discussions. Firstly, the performance of scene graph models are sensitive to the quality of the front-end detector. When the detector misses some objects, the relationships will be missed, too. Another problem is the serious imbalance in VG dataset, which makes it hard to improve the understanding of semantic relationships. It may be alleviated by utilizing external language priors.

Acknowledgements. This work is partially supported by 973 Program under contract No. 2015CB351802, Natural Science Foundation of China under contracts Nos. 61390511, 61772500, CAS Frontier Science Key Research Project No. QYZDJ-SSWJSC009, and Youth Innovation Promotion Association No. 2015085.

References

[1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. Tensorflow: a system for large-scale machine learning. In *USENIX Symposium on Operating Systems Design and Implementation (OSDI)*, volume 16, pages 265–283, 2016. 6

[2] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. Lawrence Zitnick, and D. Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2425–2433, 2015. 1

[3] S. Belongie, J. Malik, and J. Puzicha. Shape context: A new descriptor for shape matching and object recognition. In *Advances in Neural Information Processing Systems (NIPS)*, pages 831–837, 2001. 3

[4] X. Chen and A. Gupta. Spatial memory for context reasoning in object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 4106–4116, 2017. 2, 3

[5] X. Chen, L.-J. Li, L. Fei-Fei, and A. Gupta. Iterative visual reasoning beyond convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7239–7248, 2018. 2, 3, 5

[6] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014. 4

[7] B. Dai, Y. Zhang, and D. Lin. Detecting visual relationships with deep relational networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3298–3308, 2017. 1, 2, 3, 7

[8] S. K. Divvala, D. Hoiem, J. H. Hays, A. A. Efros, and M. Hebert. An empirical study of context in object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1271–1278, 2009. 3

[9] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth. Describing objects by their attributes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1778–1785, 2009. 3

[10] R. Girshick. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1440–1448, 2015. 3

[11] G. Heitz and D. Koller. Learning spatial context: Using stuff to find things. In *Proceedings of European Conference on Computer Vision (ECCV)*, volume 5302, pages 30–43. Springer, 2008. 3

[12] D. Hoiem, A. A. Efros, and M. Hebert. Geometric context from a single image. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 654–661, 2005. 3

[13] J. Johnson, A. Gupta, and L. Fei-Fei. Image generation from scene graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1219–1228, 2018. 1

[14] J. Johnson, R. Krishna, M. Stark, L.-J. Li, D. Shamma, M. Bernstein, and L. Fei-Fei. Image retrieval using scene graphs. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3668–3678, 2015. 2

[15] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. S. Bernstein, and L. Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision (IJCV)*, 123(1):32–73, 2017. 2

[16] L. Ladicky, C. Russell, P. Kohli, and P. H. Torr. Graph cut based inference with co-occurrence statistics. In *Proceedings of European Conference on Computer Vision (ECCV)*, volume 6315, pages 239–253. Springer, 2010. 3

[17] L.-J. Li, H. Su, L. Fei-Fei, and E. P. Xing. Object bank: A high-level image representation for scene classification & semantic feature sparsification. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1378–1386, 2010. 3

[18] Y. Li, W. Ouyang, X. Wang, and X. Tang. Vip-cnn: Visual phrase guided convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7244–7253, 2017. 1, 2, 3

[19] Y. Li, W. Ouyang, B. Zhou, J. Shi, C. Zhang, and X. Wang. Factorizable net: an efficient subgraph-based framework for scene graph generation. In *Proceedings of European Conference on Computer Vision (ECCV)*, volume 11205, pages 346–363. Springer, 2018. 1, 2

[20] Y. Li, W. Ouyang, B. Zhou, K. Wang, and X. Wang. Scene graph generation from objects, phrases and region captions. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1261–1270, 2017. 1, 2, 6, 7

[21] X. Liang, L. Lee, and E. P. Xing. Deep variation-structured reinforcement learning for visual relationship and attribute detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4408–4417, 2017. 1, 2

[22] Y. Liu, R. Wang, S. Shan, and X. Chen. Structure inference net: Object detection using scene-level context and instance-level relationships. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6985–6994, 2018. 2, 3

[23] C. Lu, R. Krishna, M. Bernstein, and L. Fei-Fei. Visual relationship detection with language priors. In *Proceedings of European Conference on Computer Vision (ECCV)*, volume 9905, pages 852–869. Springer, 2016. 1, 2, 5, 6, 7

[24] J. Lu, J. Yang, D. Batra, and D. Parikh. Neural baby talk. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7219–7228, 2018. 1

[25] M. Marszalek, I. Laptev, and C. Schmid. Actions in context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2929–2936, 2009. 3

[26] V. Mnih, N. Heess, A. Graves, et al. Recurrent models of visual attention. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2204–2212, 2014. 6

[27] A. Newell and J. Deng. Pixels to graphs by associative embedding. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2171–2180, 2017. 1, 2, 6, 7

[28] A. Oliva and A. Torralba. The role of context in object recognition. *Trends in cognitive sciences*, 11(12):520–527, 2007. 3

[29] D. Parikh and K. Grauman. Relative attributes. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 503–510, 2011. 3

[30] D. Parikh, C. L. Zitnick, and T. Chen. From appearance to context-based recognition: Dense labeling in small images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8, 2008. 3

[31] J. Peyre, I. Laptev, C. Schmid, and J. Sivic. Weakly-supervised learning of visual relations. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 5179–5188, 2017. 2

[32] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie. Objects in context. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1–8, 2007. 3

[33] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 91–99, 2015. 3, 5

[34] J. Shotton, J. Winn, C. Rother, and A. Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *Proceedings of European Conference on Computer Vision (ECCV)*, volume 3951, pages 1–15. Springer, 2006. 3

[35] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 5

[36] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin. Context-based vision system for place and object recognition. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 273–280, 2003. 3

[37] A. Torralba and P. Sinha. Statistical context priming for object detection. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 763–770, 2001. 3

[38] P. Wang, Q. Wu, C. Shen, A. Dick, and A. van den Hengel. Fvqa: Fact-based visual question answering. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 40(10):2413–2427, 2018. 1

[39] P. Wang, Q. Wu, C. Shen, and A. van den Hengel. The vqa-machine: Learning how to use existing vision algorithms to answer new questions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3909–3918, 2017. 1

[40] Q. Wu, C. Shen, P. Wang, A. Dick, and A. van den Hengel. Image captioning and visual question answering based on attributes and external knowledge. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 40(6):1367–1381, 2018. 1

[41] D. Xu, Y. Zhu, C. B. Choy, and L. Fei-Fei. Scene graph generation by iterative message passing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5410–5419, 2017. 1, 2, 3, 5, 6, 7

[42] J. Yang, J. Lu, S. Lee, D. Batra, and D. Parikh. Graph r-cnn for scene graph generation. In *Proceedings of European Conference on Computer Vision (ECCV)*, volume 11205, pages 690–706. Springer, 2018. 1, 2, 6, 7

[43] X. Yang, H. Zhang, and J. Cai. Shuffle-then-assemble: Learning object-agnostic visual relationship features. In *Proceedings of European Conference on Computer Vision (ECCV)*, volume 11216, pages 38–54, 2018. 1, 2

[44] B. Yao and L. Fei-Fei. Modeling mutual context of object and human pose in human-object interaction activities. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17–24, 2010. 3

[45] T. Yao, Y. Pan, Y. Li, and T. Mei. Exploring visual relationship for image captioning. In *Proceedings of European Conference on Computer Vision (ECCV)*, volume 11218, pages 711–727. Springer, 2018. 1

[46] G. Yin, L. Sheng, B. Liu, N. Yu, X. Wang, J. Shao, and C. C. Loy. Zoom-net: Mining deep feature interactions for visual relationship recognition. In *Proceedings of European Conference on Computer Vision (ECCV)*, volume 11207, pages 330–347. Springer, 2018. 4

[47] R. Yu, A. Li, V. I. Morariu, and L. S. Davis. Visual relationship detection with internal and external linguistic knowledge distillation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1974–1982, 2017. 2

[48] R. Zellers, M. Yatskar, S. Thomson, and Y. Choi. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5831–5840, 2018. 1, 2, 3, 6, 7

[49] H. Zhang, Z. Kyaw, S.-F. Chang, and T.-S. Chua. Visual translation embedding network for visual relation detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5532–5540, 2017. 1, 2, 3

[50] H. Zhang, Z. Kyaw, J. Yu, and S.-F. Chang. Ppr-fcn: Weakly supervised visual relation detection via parallel pairwise r-fcn. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 4233–4241, 2017. 2

[51] J. Zhang, M. Elhoseiny, S. Cohen, W. Chang, and A. M. Elgammal. Relationship proposal networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5678–5686, 2017. 1, 2