Composite Relationship Fields with Transformers for Scene Graph Generation

George Adaimi, David Mizrahi, Alexandre Alahi
EPFL, VITA
firstname.lastname@epfl.ch

Abstract

Scene graph generation (SGG) methods extract relationships between objects. While most methods focus on improving top-down approaches, which build a scene graph based on detected objects from an off-the-shelf object detector, there is a limited amount of work on bottom-up approaches, which jointly detect objects and their relationships in a single stage.

In this work, we present a novel bottom-up SGG approach by representing relationships using Composite Relationship Fields (CoRF). CoRF turns relationship detection into a dense regression and classification task, where each cell of the output feature map identifies surrounding objects and their relationships. Furthermore, we propose a refinement head that leverages Transformers for global scene reasoning, resulting in more meaningful relationship predictions. By combining both contributions, our method outperforms previous bottom-up methods on the Visual Genome dataset by 26% while preserving real-time performance.

1. Introduction

Human perception goes beyond object detection and recognition. It involves understanding the relations and context of a visual scene. A complete representation of an image can be described using a scene graph: a structure that extracts relationships between objects and contains rich semantic information about the scene. Extracting this structure is known as Scene Graph Generation (SGG). This structure has proven to be an effective representation for different computer vision tasks such as image captioning [85, 86], visual question answering [24], action recognition [23], and scene synthesis [10, 25]. In a scene graph, objects are represented as nodes, and the relationships between them are modeled as directed edges between the nodes. Each edge and its two corresponding nodes can also be represented by a triplet <*subject, predicate, object*>. Both the subject and object are described by a bounding box and specific object category, while the predicate is only described by the category.

Figure 1: **Our proposed Composite Relationship Fields.** We present a bottom-up scene graph generation method that leverages Composite Relationship Fields (CoRF) to produce meaningful relationships. The CoRF visualized in the figure represent the relationship *looking at* indicating that the person is *looking at* the sign.

Researchers have typically tackled this problem with top-down approaches [84, 12] that require a two-stage pipeline: (1) an object detector that detects the different objects in the scene [59], from which a fully-connected scene graph is built, and (2) a relationship predictor that predicts the relationship edge between every pair of objects. Since objects and predicates are not predicted simultaneously, and relationship prediction is performed for every pair of objects, the computation cost increases quadratically with the number of objects. This renders top-down approaches inefficient with low inference speed. Given that SGG is not necessarily the final task and the scene graphs are often used as a semantically rich intermediate representation for other downstream tasks[23, 46], the inference speed of the downstream task is bounded by the low speed of top-down SGG methods.

Recently, there have been inspiring bottom-up approaches [55, 47] aiming to address the above limitation by predicting the complete scene graph in one step. They eliminate the need to build a fully-connected graph by di-

rectly predicting the relationships and objects in a single step. They are based on the success of keypoint-based methods for human pose detection. Since both object and predicate detection are performed on the whole input image, bottom-up methods can encode contextual information from the whole scene. Furthermore, predicting the scene graph in a single step enables a more efficient method for SGG, which is critical for accelerating downstream tasks that benefit from a semantically meaningful representation of a scene. However, the gap in accuracy between top-down and bottom-up methods remains high. In this work, our goal is to reduce the gap by proposing a new bottom-up method that outperforms existing bottom-up approaches on Visual Genome by 26% on scene graph detection (SGDet), paving the way to real-time execution of downstream tasks that rely on SGG.

One crucial observation pertaining to scene-graph construction is that the subject and object are highly indicative of their corresponding relationship [93]. For example, knowing that the subject is a 'person' and the object is a 'horse', it is highly probable that the relationship is 'ride' and not 'wear'. This shows that a correlation exists between the objects and their relationships. To leverage this correlation, top-down approaches include different refinement techniques that use the information extracted from the object detector, such as object categories and bounding boxes, to improve relationship prediction[81, 38]. Existing bottom-up methods do not explicitly leverage such correlation since both objects and relationships are predicted simultaneously rather than sequentially. Although bottom-up methods can benefit from contextual information as they can reason on the whole scene, current approaches [55, 47] rely on convolutional neural networks (CNNs) that exhibit small effective receptive fields [51]. By analyzing the contribution of each input pixel to the receptive field of a CNN's output, it has been shown that a specific feature's response is strongest at the center of its receptive field and decays quickly the farther the input signal is from its center [51]. This means that a specific output location might not be able to capture information that is far away, and thus, previous bottom-up models can suffer when relating objects that are far away from each other.

In this paper, we address the limitations of bottom-up approaches to improve scene graph generation while maintaining their main advantage, efficiency. Inspired by association fields [30], we first introduce a novel bottom-up approach that uses *Composite Relationship Fields (CoRF)* to represent relationships between objects and generates a scene graph in a single forward pass. We then propose a new refinement head for bottom-up SGG methods that leverages the Transformer architecture [72] to reason about all objects in the scene while benefiting from the global image. This provides a mechanism for bottom-up methods to leverage the correlation that exists between the objects and their relationships.

Specifically, we present the following contributions:

- We show that Composite Relationship Fields (CoRF) can effectively represent relationships between objects, leading to richer relationship representations and a significant boost in relationship detection.
- We propose a new Transformer-based refinement head for bottom-up SGG methods that enables global scene reasoning, further improving relationship prediction.

Our method, Composite Relationship Fields with a Transformer-based refinement head, outperforms the state-of-the-art bottom-up methods [47]. It also shows strong generalization capabilities, outperforming existing bottom-up methods on zero-shot recall by 50%. Code is available at https://github.com/vita-epfl/SGG-CoRF.

2. Related Work

Top-down methods for SGG. Top-down methods are methods that first use an object detector or region proposal network (RPN) [59] to detect the objects in the scene and build an initial fully-connected graph where each edge refers to a relationship. Then, a representation of each edge is used to predict the relationship or even if a relationship exists. These methods mostly focus on refining the representations of the relationships and objects using iterative message passing [81, 7, 77] and graph neural networks [12, 65, 44]. Subsequent methods further introduce other refinement techniques to include global context of the scene [78]. Some methods focus on reducing the number of initial edges to improve efficiency [84, 50] while others focus on dealing with the biased distribution of scene graph datasets [36, 83, 34, 45].

While the above methods reason on visual information only, including image segmentation [26], there are also a range of top-down methods that leverage extra information, *e.g.*, from language or knowledge graphs [15, 57, 89, 62, 18, 95, 87, 98, 19, 88, 90, 91, 21, 49, 68, 37, 11] or from prior dataset statistics [93, 96, 63, 52, 17]. GPS-Net [43] addresses the direction of the relationship and the long-tailed distribution of relations. Similarly, the Knowledge-Embedded Routing Network (KERN) [5] creates a knowledge graph and uses message passing to address the predicate class imbalance. MotifsTDE [67] devises a method that handles biased training data using Total Direct Effect (TDE). Other frameworks of SGG produce different types of scene graphs such as 3D scene graphs [2, 79, 94, 73], dynamic scene graphs [6, 71, 39] and topic scene graphs [76].

Bottom-up methods for SGG. In recent work, FC-SGG [47] extends the Parts Affinity Fields (PAF) introduced in OpenPose [3] to encode the relationships between objects. Our approach extends the association fields [30] that have shown better performance than PAFs in keypoint estimation due to their high precision regression as well as their ability to predict associations between overlapping instances. The latter is crucial for scene graph generation since some objects

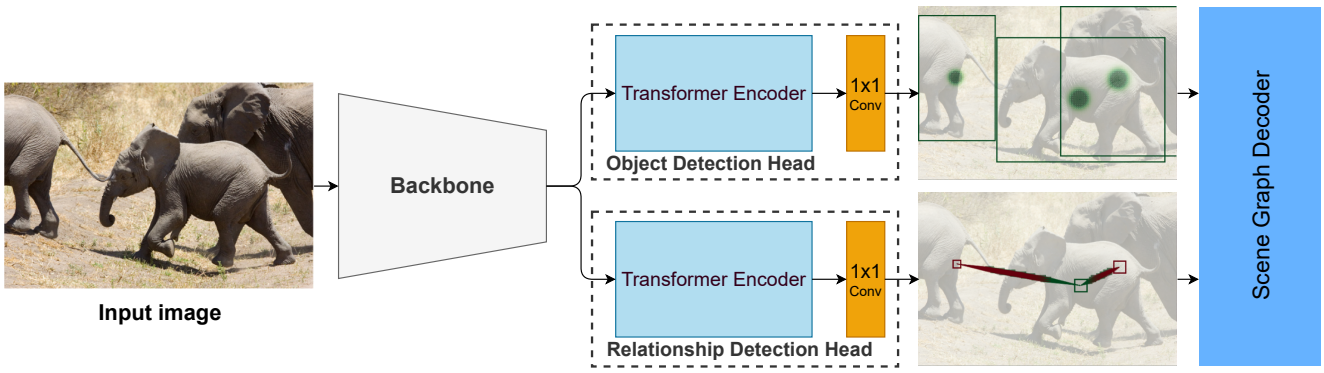


Figure 2: **Model overview**. Our method uses Transformers to refine the feature map from a backbone for object and relationship detection. Relationship detection is done using Composite Relationship Fields (CoRF). The decoder takes as input the detections and relationships to form the complete scene graph.

such as "eyes" and "head", or "person" and "shirt" might overlap at the same output position. The association between such close objects will not be detected by FCSGG. Pixels to Graphs by Associative Embedding (Px2Graph) [55] is another bottom-up method that uses a generic embedding space to form associations between the predicate and the subject and object. Px2Graph, and other one-stage methods [35, 70], are still unable to run in real-time which prevents them from being used for down-stream applications.

Bottom-up methods for Human-Object Interaction. Human-object interaction (HOI) [27, 16] is a closely-related task to SGG where the subject of a relationship is a human and the relationship is restricted to interactions (*e.g.,* riding, holding). Certain bottom-up strategies in HOI [40, 75] predict the humans and objects as keypoints in the scene in addition to a vector indicating the association between them. Since SGG contains more types of relationships (*e.g.,* spatial, interaction) and between any two objects, we make use of the denser CoRF which enables us to reason on different features in the scene for more complex relationships.

Transformers in vision The Transformer architecture[72] first proposed for sequence-to-sequence translation has recently seen many applications in computer vision. In particular, recent works use it either as a standalone backbone [13] or in a hybrid CNN + Transformer architecture to refine feature maps [4, 100, 60]. Such architectures have been applied to many visual tasks, such as classification [13, 48], detection [4, 14], semantic segmentation [58, 97], tracking [53], pose estimation [33] and human-object interaction [101, 28].

3. Method

This work focuses on improving relationship prediction between various objects in the scene. To this end, we propose a computationally efficient bottom-up method for SGG based on composite fields for relationship prediction.

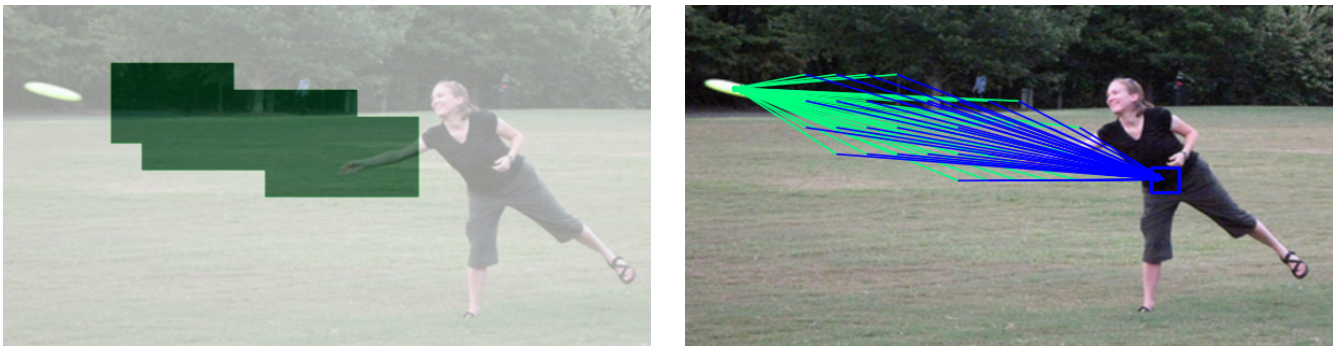
Our overall architecture is shown in Figure 2. It consists of (1) a backbone network that extracts a feature map, (2) an object detection head, and (3) a relationship detection head. Each head consists of a Transformer encoder followed by a 1×1 convolutional layer. More specifically, let $\mathcal{I} \in \mathbb{R}^{3 \times H \times W}$ be the input image of width W and height H. The image is passed to the backbone network (*e.g.*, ResNet-50) and a feature map $F \in \mathbb{R}^{C \times \frac{H}{s} \times \frac{W}{s}}$ is extracted, where C is the number of channels and s is the output stride, set to 16 in our experiments. The extracted feature map is then passed to the object and relationship detection heads.

3.1. Object Detection

We employ the one-stage anchor-free CenterNet [99] as the object detector, similar to FCSGG [47]. CenterNet has shown high performance on the widely-used MSCOCO [42] object detection dataset. Given feature map F extracted from the backbone, the object detection head outputs three feature maps: (1) a heatmap $\hat{H}^o$ indicating the centers and categories of the objects, (2) a center offset to deal with the precision error caused by the output stride, and (3) the objects' width and height. Details about CenterNet and its implementation can be found in the appendix.

3.2. Composite Fields for Relationship Detection

The main challenge in bottom-up methods for scene graph generation is to directly group objects that have a relationship. Px2Graph [55] uses associative embeddings to obtain these groupings, while FCSGG [47] uses affinity fields inspired from Parts Affinity Fields [3]. Motivated by the success of composite fields introduced by Kreiss *et al.* [29] and their applicability to different tasks such as keypoint estimation and tracking [30, 92] and attribute detection [54], we introduce Composite Relationship Fields (CoRF) to detect relationships between objects, illustrated in Figure 3. For each cell of the relationship head's feature map, CoRF



(a) (b)

Figure 3: **Illustration of the CoRF for the predicate *throwing***: (a) shows the confidence of the activated region and (b) shows the vectors originating from each activated cell. The blue and green vectors are v_s and v_o pointing to the subject and object, respectively. The confidence score is high in the area between the two related objects.

predicts the existence of a relationship as well as the position of the subjects and objects involved if the cell is situated between the subject and object. To do so, for each possible predicate, the relationship head outputs a confidence score, two vectors pointing to the subject and the object, respectively, and a scale for both the subject and the object. Thus, at every location (i, j) and channel p of the feature map outputted by the relationship head, CoRF can be represented by: $a^p_{ij} = [c, x_s, y_s, x_o, y_o, s_s, s_o]^p_{ij}$ where c is the confidence for predicate p, (x_s, y_s) and (x_o, y_o) are the vector components indicating the position of the subject and object relative to position (i, j) in the feature map, and s_s and s_o are the scales of the subject and object. The scale is specified as one-tenth the minimum of the width and height of the corresponding bounding box. Therefore, the relationship head outputs a vector of size $7 \cdot |p|$ for each cell of the feature map, with $|p|$ the number of possible predicates. As a result, our model predicts multiple relationships at the same location.

3.3. Feature Refinement with Transformers

When using CoRF, each cell of the feature map has to identify objects around it and then reason about the relationships between them. Cells must also to determine their relative position to these objects in order to point to them. This task is further complicated by the fact that some relationships span the entire image, which requires the cell to identify far-away objects. For these reasons, we propose using a Transformer encoder [72] to refine features as it has several desirable properties: (1) With multi-headed self-attention, each cell can query its surroundings and attend to multiple objects at once. (2) Positional encodings can be used to determine the relative position between objects and the cells. (3) Unlike convolutional layers, self-attention layers are global and can thus attend to objects regardless of their distance from the cell.

Our refinement head works in the following way: we project the feature map to the Transformer dimension d and then flatten it. Next, we add positional encodings to the flattened features to preserve spatial information before feeding it into the Transformer. Finally, we reshape the output tokens into an image-like map which is passed to a 1×1 convolutional layer to obtain the final prediction.

As Transformers have been shown to be effective at refining features for object detection [4, 66], we add an identical Transformer encoder to the object detection head.

Transformer architecture. We use a Transformer encoder with six layers of width $d = 256$, each with eight attention heads and a pointwise MLP with one hidden layer of width $8d$. This encoder has been shown to be effective at refining features while still being lightweight [4].

Positional encodings. We use fixed sinusoidal positional encodings similar to [4, 1], to which we concatenate the absolute position values. This positional encoding is then projected to the Transformer dimension d and added to the input features. We did not observe any performance gains from using learnable 1D and 2D positional encodings.

3.4. Training Supervision

The confidence heatmaps are trained using a modified focal loss, while the vectors, offsets, box size, and scales are trained using l_1 loss. The focal loss used is a variant that modulates the loss based on the ground-truth heatmap [32]. For a ground-truth heatmap H and a predicted heatmap $\hat{H}$, the modified focal loss is as follows:

$$L_{Focal} = -\frac{1}{N} \sum_{c,y,x} \begin{cases} (1 - \hat{H}_{c,y,x})^{\alpha} \log(\hat{H}_{c,y,x}) & \text{if } H_{c,y,x} = 1 \\ (1 - H_{c,y,x})^{\beta} \, \hat{H}^{\alpha}_{c,y,x} \, \log(1 - \hat{H}_{c,y,x}) & \text{otherwise.} \end{cases}$$

where N is the number of activated feature cells in the heatmap, and α and β are hyperparameters set to 2 and 4, respectively. The remaining output maps are trained using

an l_1 loss. The final loss is the equal sum of all the losses from both the detection and relationship head.

3.5. Scene Graph Decoding

To build the scene graph, we first extract the top 100 objects predicted by our object detector head. Extraction is done by performing a 3×3 max-pooling operation over the predicted heatmap $\hat{H}^o$ to extract the top 100 peaks. The location of peaks indicates the center of an object. It also indicates the detection confidence for the category of the object. At the location of every peak, we obtain both the offset and bounding box size (width and height) from the feature maps predicted by the object detection head. The offset is used to restore the precision error lost during downsampling.

The second step extracts the relationships between the detected objects using the CoRF predicted by our relationship head. We first extract the relationships with a confidence score higher than a specific threshold τ, *i.e.*, only the fields with $c > \tau = 0.1$ are considered. Given the detected objects, the next step is to identify which objects form the subject and object of a relationship, *i.e.*, finding the closest object to each of the CoRF subject (x_s, y_s) and object vectors (x_o, y_o). This is done by computing a weighted ℓ_2 norm between the location of every detected object and every predicted CoRF vector. For instance, the object that is located closest to the subject vector of a specific relationship is considered as the *subject* of the relationship. More specifically, at output position (i, j), the ℓ_2 norm between object b at center position (x_c, y_c) and CoRF subject vector (x_s, y_s) or object vector (x_o, y_o) is weighted by the confidence score c as follows:

$$\ell_{2(r_{ij},b)} = \frac{1}{c} \left\| (x_c, y_c) - [(i, j) + (x_{s/o}, y_{s/o})] \right\|_2 .$$

The ℓ_2-norm decreases with higher c to favor objects with higher confidence scores. We add (i, j) to the CoRF vector components $(x_{s/o}, y_{s/o})$ in the above equation to convert them from relative to absolute position. Although our method allows relationships between the same object, we eliminate such cases since there is no such scenario in the Visual Genome [31] dataset.

3.6. Discussion

RAF vs. CoRF. Composite Relationship Fields are used to associate the subject and object involved in a relationship. Compared to the Relationship Affinity Fields (RAF) used by FCSGG [47], CoRF does not suffer from discretization error since it predicts a real-value vector pointing towards the components in a relationship. To illustrate, Visual Genome (VG) has 150 object categories, including the 'nose', 'hair', and 'head' category. The center of these categories can be very close in the image. Since the backbone downsamples the input image, the center of these categories might overlap at a specific cell in the grid-based output feature map. If a relationship 'on' exists between the 'hair' and 'head' in an image, RAF will point from the 'hair' center to the cell where both the center of the head and nose lie. This leads to a wrong relationship prediction between 'hair' and 'nose' (<hair, on, nose>). CoRFs can accurately point to either the nose or head since the vector does not point to a specific cell but to a real-value point which is the object's exact position in the input image. Consequently, CoRF allows association between objects in the same output cell, such as the nose and head. This enables CoRF to work with low-resolution feature maps.

4. Experiments

4.1. Dataset & Training Details

Visual Genome dataset. We evaluate our method on Visual Genome (VG) [31], a publicly-used dataset for scene graph generation. We use the most common preprocessed subset of VG-150 [81], which includes the most frequent 150 object categories and 50 predicate categories.

Technical details. We report results with a convolutional backbone: ResNet-50 [20], and a Transformer backbone: Swin-S [48]. These backbones are pretrained on ImageNet [61] and modified to output a feature map of stride 16. Details about the architecture and training procedure are provided in the appendix.

4.2. Evaluation

We follow the standard evaluation protocols used to measure the performance of an SGG method:

- Scene Graph Detection (**SGDet**): detect object bounding boxes, categories, and relationships.
- Scene Graph Classification (**SGCls**): given object bounding boxes, detect the categories and relationships.
- Predicate Classification (**PredCls**): given object bounding boxes and categories, detect relationships.

Since the ground-truth annotations are incomplete, Recall@K is adopted as the evaluation metric, where only the top K relationship predictions are considered. We also report the results following the No-Graph Constraint setting (ngR@K) [93], where multiple relationships can exist per object pair. We further report the mean recall@K (mR@K) [5, 69] since the distribution of relationships has a long tail. To study the generalization capabilities of our model, we also report the zero-shot performance, which evaluates the recall on subject-object relationships not found in the training set.

In PredCls and SGCls settings, the relationship head should only be evaluated for its ability to classify the relationship and both object and relationship, respectively. Our relationship head not only predicts the categories of relationships but also the locations of the subject and object,

	Method	Backbone	GT	$AP_{0.5}$	PredCls		SGCls		SGDet		img/sec
					R@50	ng-R@50	R@50	ng-R@50	R@50	ng-R@50	
Top-down	Graph R-CNN [84]	VGG16 [64]	✓	23.0	54.2	–	29.6	–	11.4	–	5*
	Seq2Seq [50]	VGG16 [64]	✓	–	66.4	83.6	38.3	46.9	30.9	30.9	–
	BGNN [36]	RNXt101-FPN	✓	–	59.2	–	37.4	–	31.0	–	1.8
Bottom-up	Px2Graph [55]	Hourglass-104 [56]	✓	–	54.1	68.0	21.8	26.5	8.1	9.7	0.2
	CoRF + T^+ [Ours]	RN50	✓	21.9	60.5	78.8	28.6	36.1	16.5	20.2	22
	CoRF + T^+ [Ours]	Swin-S	✓	24.7	60.2	78.5	30.5	38.8	18.6	22.9	15
Bottom-up	FCSGG [47]	HRNet-W48 [74]	✗	25.0	31.0	40.3	17.1	19.6	15.5	18.3	13
	FCSGG [47]	HRNet-W32 [74]	✗	21.6	34.9	46.3	15.5	19.3	15.1	18.2	14
	FCSGG [47]	RN50-$FPN_{\times 2}$	✗	23.0	35.8	44.7	17.7	20.6	15.7	18.0	25
	CoRF [Ours]	RN50	✗	19.6	42.3	53.9	14.8	18.3	14.5	17.6	**30**
	CoRF + T [Ours]	RN50	✗	21.9	44.4	56.8	17.2	21.3	16.5	20.2	22
	CoRF [Ours]	Swin-S	✗	23.8	44.8	56.9	17.5	21.6	17.9	22.0	20
	CoRF + T [Ours]	Swin-S	✗	24.7	**45.4**	**58.1**	**18.7**	**23.4**	**18.6**	**22.9**	15

Table 1: **Recall@50 for graph (R@50) and no-graph constraint (ng-R@50) on Visual Genome [31]** The top section shows the performance of methods that use ground-truth information in the model under the PredCls and SGCls protocol. Methods in the bottom section do not use ground-truth information. GT: ground-truth detections used in the model under PredCls and SGCls. RN50 = ResNet50[20]. RNXt101 = ResNeXt101[80]. FPN = Feature Pyramid Network [41]. $FPN_{\times 2}$: separate FPNs are used for detection and relationship. $^+$: GT detections given as tokens to the Transformer. We report the best performing top-down visual only method (Seq2Seq) and a few others for compactness. A complete table is included in the appendix. CoRF + T has a Transformer encoder in both heads. Inference speed is measured on a Nvidia GeForce GTX 1080 Ti GPU, with images of resolution 512×512 and a batch size of 1 under SGDet protocol. *: input resolution is 800×1024.

represented by the CoRF vector components. To compare with previous work, especially top-down methods, we need to provide our model with ground-truth object annotations. We do this by providing ground-truth (GT) detections as additional tokens to the Transformer modules in both the detection and relationship head. For the SGCls protocol, only the location and size of the objects are provided, while for PredCls, we provide the location, size, and category. Details on the implementation are in the appendix.

5. Results

Recall performance. We report Recall@50 performance in Table 1. The top section of the table shows the performance of methods that use ground-truth annotations in the model under the PredCls and SGCls protocol, while the bottom section includes models that do not. We further divide the state-of-the-art models into bottom-up and top-down approaches. We report the performance of methods that are relevant for comparison and only rely on visual information, similar to our method. A complete table with other top-down methods that also use external knowledge is available in the appendix. The complete table also reports results for Recall@20 and Recall@100 showing similar improvements.

As observed in the bottom section of Table 1, our method, Composite Relationship Fields with Transformers (CoRF + T), significantly improves performance on both ResNet-50 and Swin-S. First, we compare it to the state-of-the-art bottom-up method, FCSGG with HRNet-W32, and show our performance is 23% and 26% better for PredCls ng-R@50 on ResNet-50 and Swin-S, respectively. This shows that our relationship head with the Transformer refinement learns better representations for relationship detection. Our method and FCSGG do not use ground-truth annotations under PredCls and SGCls. Moreover, our method achieves a gain of 10% (ResNet-50) and 26% (Swin-S) on SGDet ng-R@50 compared to FCSGG.

Since Px2Graph passes GT annotations to the model as input when reporting performance under PredCls and SGCls (Section 4.2), we also pass these annotations as tokens to the Transformer module to fairly compare to it (top section of Table 1). We outperform it by a large margin on the three protocols going from 68.0 to 78.5 on PredCls (ng-R@50) and 9.7 to 22.9 on SGDet (ng-R@50).

As shown in the top section of Table 1, our model (CoRF + T^+) is on par with and even outperforms certain top-down methods on PredCls while running in real-time. It is important to note that the relationship head of top-down methods only needs to perform classification without any regression task since the objects are initially extracted, and the prediction is performed for every pair of objects. Performing such pairwise prediction simplifies the role of the head but is the main reason why top-down methods are inefficient and cannot be used in real-time applications. On the other

	Method	Backbone	PredCls mR/ng-mR	SGCls mR/ng-mR	SGDet mR/ng-mR
Top-down	KERN [5]	VGG16	17.7/–	9.4/–	6.4/–
	MOTIFS-TDE [67]	RNXt101-FPN	25.5/–	13.1/–	8.2/–
	Seq2Seq [50]	VGG16	26.1/–	14.7/–	9.6/–
	PCPL [82]	VGG16	35.2/50.6	18.6/26.8	9.5/10.4
	DT2-ACBS [9]	RN101-FPN	35.9/–	24.8/–	22.0/–
Bottom-up	FCSGG	HRNet-W32	5.5/9.7	2.5/4.4	2.4/3.6
	FCSGG	HRNet-W48	5.2/9.5	2.9/6.3	2.6/4.7
	FCSGG	RN50-FPN$_{\times 2}$	5.7/11.3	2.9/6.0	2.7/4.9
	CoRF	RN50	8.1/17.0	2.7/5.4	2.7/5.8
	CoRF + T	RN50	9.5/20.0	3.4/6.8	3.5/7.6
	CoRF	Swin-S	9.3/19.2	3.3/6.9	3.5/7.9
	CoRF + T	Swin-S	**10.1/21.7**	**3.9/8.3**	**3.9/9.2**

Table 2: **Mean recall performance.** We compare mean recall@50 for graph (mR) and no-graph (ng-mR) constraint on Visual Genome [31]. CoRF has convolutions in both heads. CoRF + T has a Transformer encoder in both heads.

	Method	Backbone	PredCls zsR/ng-zsR	SGCls zsR/ng-zsR	SGDet zsR/ng-zsR
Top-down	VTransE-TDE [67]	RNXt101-FPN	13.3/–	2.9/–	2.0/–
	Motifs-TDE [67]	RNXt101-FPN	14.4/–	3.4/–	2.3/–
	VCTree-TDE [67]	RNXt101-FPN	14.3/–	3.2/–	2.6/–
	VCTree-TDE-EB [65]	RNXt101-FPN	15.1/–	6.4/–	2.7/–
Bottom-up	FCSGG	RN50-FPN$_{\times 2}$	8.2/11.7	1.3/2.4	0.8/1.0
	FCSGG	HRNet-W32	8.3/12.9	1.0/2.3	0.6 /1.2
	FCSGG	HRNet-W48	8.6/12.8	1.7/ 2.9	1.0/1.8
	CoRF	RN50	10.5/16.3	1.5/3.2	0.4/1.1
	CoRF + T	RN50	11.6/18.2	1.8/4.0	0.8/1.4
	CoRF	Swin-S	11.1/18.0	1.9/**3.5**	1.1/2.2
	CoRF + T	Swin-S	**11.3/18.8**	**1.9**/3.8	**1.2/2.6**

Table 3: **Zero-shot performance.** We compare zero-shot recall@50 for graph (zR) and no-graph (ng-zR) constraint on Visual Genome [31]. CoRF has convolutions in both heads. CoRF + T has a Transformer encoder in both heads.

hand, the relationship head of bottom-up methods needs not only to classify the type of relationship between objects but also to output a representation indicating which objects are related, similar to our CoRF or Px2Graph's associative embeddings [55]. Nevertheless, our model successfully reduces the gap between bottom-up and top-down methods without compromising efficiency. We note that improving the detection performance ($AP_{0.5}$) by adding a Transformer to the object head leads to further improvements in both SGCls and SGDet. Nonetheless, assuming perfect object detection, the significant improvements in PredCls compared to other bottom-up methods show that our model, specifically our relationship head, is able to encode relationships better.

Mean recall performance. We report the mean recall results in Table 2. We significantly outperform previous bottom-up approaches under all metrics, especially for PredCls. Our best model obtains a 77% and 92% improvement on mR@50 and ng-mR@50 PredCls, respectively, compared to FCSGG [47]. Similarly, we achieve a 44% and 88% increase in SGDet performance. These results show that our model can better deal with the long-tail distribution of the dataset and still output relationships that do not frequently appear in the training set. Our model better uses visual features to understand the relationship between objects instead of memorizing the most common relationship triplets. A complete table with top-down methods is included in the appendix. It is important to note that top-down methods in Table 2 apply techniques such as sampling strategies, external knowledge, etc., to specifically improve mean recall. These strategies aid in distinguishing between visually close relationships, such as *laying on* and *lying on*. These relationships are challenging for methods relying on visual-only input, similar to our method and FCSGG [47].

Zero-shot performance. We report zero-shot performance in Table 3. The results further motivate our method for its improved generalization capabilities. CoRF paired with Transformers allows the model to attend to and relate objects it has not observed during training. Our method on Swin-S leads to a 20% and 44% increase in SGDet for graph and no-graph constraint, respectively, compared to FCSGG. Previous top-down methods apply a debiasing technique (TDE [67]) to improve their zero-shot metric, leading to unfair comparison. We report these numbers for completeness. A complete table is included in the appendix.

Inference speed. We show the inference speed of our model in Table 1 (rightmost column). All models are tested on a Nvidia GeForce GTX 1080 Ti GPU with an image size of 512×512. Unlike top-down methods, all our models maintain a real-time speed while improving performance compared to previous bottom-up approaches. The inference speeds of more top-down methods are reported in the appendix and show their inability to run in real time.

6. Ablation Studies

Impact of CoRF. We study the effect of CoRF by replacing the Transformer in both heads with four 3×3 convolutional layers, each separated by a Batch Normalization layer [22] and a ReLU activation. The results are reported in Table 1 (CoRF). For this ablation study, we do not provide ground-truth annotations as tokens in PredCls and SGCls. Since the fields mainly affect relationship prediction, we report PredCls and observe an improvement of around 16% in ng-R@50 with the smaller model, ResNet-50, compared to the best performing FCSGG model. A significant improvement of 26% and 50% is also observed in zero-shot and mean recall performance, respectively. This indicates that CoRF, with its denser connections, allows the model to better generalize to new relationships and is less affected by the long-tail distribution of predicates in the training set. A larger improvement is also observed when using Swin-S.

Impact of different relationship refinement heads. To verify the benefits of Transformers for feature refinement,

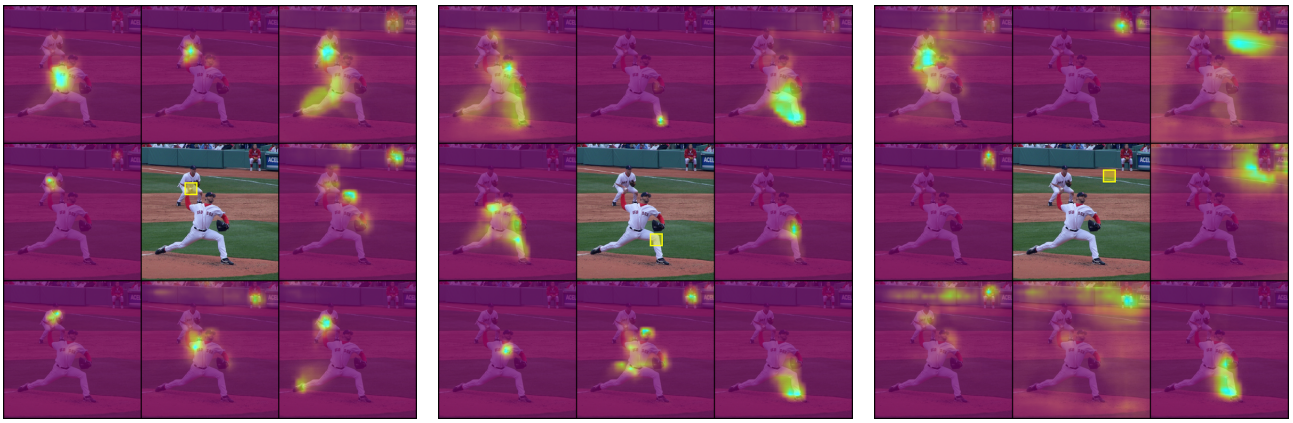


Figure 4: **Attention maps of the relationship head's Transformer.** For a given reference point (yellow box), the attention maps from **all** heads of the last self-attention layer are shown. Order of heads is arbitrary. Attention heads are able to attend to the surroundings of the cell as well as far-away objects.

		PredCls	SGCls	SGDet
Method	$AP_{0.5}$	R/ng-R	R/ng-R	R/ng-R
CoRF	19.6	42.3/53.9	14.8/18.3	14.5/17.7
CoRF + Deform	18.1	41.4/53.5	12.9/16.1	12.7/15.7
CoRF + S.Deform	19.2	41.5/53.4	14.2/17.9	14.3/17.4
CoRF + T_R	19.4	43.5/56.0	14.9/18.7	14.3/17.5
CoRF + T	**21.9**	**44.4/56.8**	**17.2/21.3**	**16.5/20.2**

Table 4: **Performance of different relationship refinement heads.** Ablation study showing the recall@50 of different relationship refinement heads with ResNet-50 for graph (R) and no-graph (ng-R) constraint. The different heads are described in Section 6 and the appendix.

we replace the Transformer encoder in the object detection head with four 3×3 convolutional layers, each separated by a Batch Normalization layer [22] and a ReLU activation, and replace the Transformer encoder in the relationship heads with different refinement heads. Table 4 shows the performance when using deformable convolutions [8] and supervised deformable convolutions, where the offsets of three deformable convolutions are trained to attend to the subject, object, and predicate of every relationship. CoRF + T_R uses a Transformer encoder only in the relationship head. The different heads are detailed in the appendix.

As observed, using a Transformer encoder only for the relationship head (CoRF + T_R) leads to a gain of $\sim 2\%$ in PredCls, indicating its benefits to improving relationship prediction. Adding the Transformer encoder to the object detection head improves $AP_{0.5}$ leading to improvements in various metrics, specifically SGCls and SGDet.

Furthermore, in Figure 4, we show the attention maps of the last self-attention layer of the relationship head, focusing on specific locations in the image (yellow box). When considering a specific point (yellow box), the attention heads focus not only on its local surroundings but also on different objects in the scene, even distant objects. These qualitative results further validate the usefulness of Transformers for relationship detection, as they are able to effectively attend to multiple objects to predict the relationships between them. We note that these attention maps are specific to the Transformer in the relationship head, as such attentions maps are different from the maps of the Transformer of the object detection head, shown in the appendix, and the Transformer of DETR [4], despite using similar architectures.

7. Conclusion

Scene graph generation enables a rich semantic and contextual understanding of a visual scene. Our work presents a novel bottom-up SGG method representing relationships as Composite Relationship Fields (CoRF). We further propose a Transformer-based refinement that can directly attend to the subjects and objects involved in a relationship. Our method outperforms other bottom-up approaches on the Visual Genome dataset. It is also on par with or even outperforms certain top-down methods while being more efficient. Our contributions also help deal with rare or even unseen relationships by the gain in mean and zero-shot recall. As our method is able to perform in real time, our scene graph representation can be leveraged to improve other real-time tasks such as action recognition and image generation.

Acknowledgement. This project has received funding from the Initiative for Media Innovation based at Media Center, EPFL, Lausanne, Switzerland

References

[1] Alaaeldin Ali, Hugo Touvron, Mathilde Caron, Piotr Bojanowski, Matthijs Douze, Armand Joulin, Ivan Laptev, Natalia Neverova, Gabriel Synnaeve, Jakob Verbeek, and Herve Jegou. Xcit: Cross-covariance image transformers. In *Advances in Neural Information Processing Systems*, volume 34, pages 20014–20027, 2021.

[2] Iro Armeni, Zhi-Yang He, JunYoung Gwak, Amir R Zamir, Martin Fischer, Jitendra Malik, and Silvio Savarese. 3d scene graph: A structure for unified semantics, 3d space, and camera. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5664–5673, 2019.

[3] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh. Openpose: Realtime multi-person 2d pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229. Springer, 2020.

[5] Tianshui Chen, Weihao Yu, Riquan Chen, and Liang Lin. Knowledge-embedded routing network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6163–6171, 2019.

[6] Yuren Cong, Wentong Liao, Hanno Ackermann, Bodo Rosenhahn, and Michael Ying Yang. Spatial-temporal transformer for dynamic scene graph generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16372–16382, October 2021.

[7] Bo Dai, Yuqi Zhang, and Dahua Lin. Detecting visual relationships with deep relational networks. In *Proceedings of the IEEE conference on computer vision and Pattern recognition*, pages 3076–3086, 2017.

[8] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[9] Alakh Desai, Tz-Ying Wu, Subarna Tripathi, and Nuno Vasconcelos. Learning of visual relations: The devil is in the tails. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15404–15413, October 2021.

[10] Helisa Dhamo, Azade Farshad, Iro Laina, Nassir Navab, Gregory D Hager, Federico Tombari, and Christian Rupprecht. Semantic image manipulation using scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5213–5222, 2020.

[11] Xingning Dong, Tian Gan, Xuemeng Song, Jianlong Wu, Yuan Cheng, and Liqiang Nie. Stacked hybrid-attention and group collaborative learning for unbiased scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19427–19436, June 2022.

[12] Apoorva Dornadula, Austin Narcomey, Ranjay Krishna, Michael Bernstein, and Fei-Fei Li. Visual relationships as functions: Enabling few-shot scene graph prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[14] Yuxin Fang, Bencheng Liao, Xinggang Wang, Jiemin Fang, Jiyang Qi, Rui Wu, Jianwei Niu, and Wenyu Liu. You only look at one sequence: Rethinking transformer in vision through object detection. In *Advances in Neural Information Processing Systems*, volume 34, pages 26183–26197, 2021.

[15] Nikolaos Gkanatsios, Vassilis Pitsikalis, Petros Koutras, and Petros Maragos. Attention-translation-relation network for scalable scene graph generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019.

[16] Georgia Gkioxari, Ross Girshick, Piotr Dollár, and Kaiming He. Detecting and recognizing human-object interactions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[17] Arushi Goel, Basura Fernando, Frank Keller, and Hakan Bilen. Not all relations are equal: Mining informative labels for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15596–15606, June 2022.

[18] Jiuxiang Gu, Handong Zhao, Zhe Lin, Sheng Li, Jianfei Cai, and Mingyang Ling. Scene graph generation with external knowledge and image reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1969–1978, 2019.

[19] Yuyu Guo, Lianli Gao, Xuanhan Wang, Yuxuan Hu, Xing Xu, Xu Lu, Heng Tao Shen, and Jingkuan Song. From general to specific: Informative scene graph generation via balance adjustment. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16383–16392, October 2021.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[21] Roei Herzig, Moshiko Raboh, Gal Chechik, Jonathan Berant, and Amir Globerson. Mapping images to scene graphs with permutation-invariant structured prediction. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[22] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015.

[23] Jingwei Ji, Ranjay Krishna, Li Fei-Fei, and Juan Carlos Niebles. Action genome: Actions as compositions of spatio-temporal scene graphs. In *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition, pages 10236–10247, 2020.

[24] Juanzi Li Jiaxin Shi, Hanwang Zhang. Explainable and explicit visual reasoning over scene graphs. In *CVPR*, 2019.

[25] Justin Johnson, Agrim Gupta, and Li Fei-Fei. Image generation from scene graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1219–1228, 2018.

[26] Siddhesh Khandelwal, Mohammed Suhail, and Leonid Sigal. Segmentation-grounded scene graph generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15879–15889, October 2021.

[27] Bumsoo Kim, Taeho Choi, Jaewoo Kang, and Hyunwoo J. Kim. Uniondet: Union-level detector towards real-time human-object interaction detection. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part XV*, volume 12360 of *Lecture Notes in Computer Science*, pages 498–514. Springer, 2020.

[28] Bumsoo Kim, Junhyun Lee, Jaewoo Kang, Eun-Sol Kim, and Hyunwoo J Kim. Hotr: End-to-end human-object interaction detection with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 74–83, 2021.

[29] Sven Kreiss, Lorenzo Bertoni, and Alexandre Alahi. Pifpaf: Composite fields for human pose estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11977–11986, 2019.

[30] Sven Kreiss, Lorenzo Bertoni, and Alexandre Alahi. Openpifpaf: Composite fields for semantic keypoint detection and spatio-temporal association. *IEEE Transactions on Intelligent Transportation Systems*, 2021.

[31] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International journal of computer vision*, 123(1):32–73, 2017.

[32] Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[33] Ke Li, Shijie Wang, Xiang Zhang, Yifan Xu, Weijian Xu, and Zhuowen Tu. Pose recognition with cascade transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1944–1953, 2021.

[34] Lin Li, Long Chen, Yifeng Huang, Zhimeng Zhang, Songyang Zhang, and Jun Xiao. The devil is in the labels: Noisy label correction for robust scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18869–18878, June 2022.

[35] Rongjie Li, Songyang Zhang, and Xuming He. Sgtr: End-to-end scene graph generation with transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19486–19496, June 2022.

[36] Rongjie Li, Songyang Zhang, Bo Wan, and Xuming He. Bipartite graph network with adaptive message passing for unbiased scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11109–11119, June 2021.

[37] Wei Li, Haiwei Zhang, Qijie Bai, Guoqing Zhao, Ning Jiang, and Xiaojie Yuan. Ppdl: Predicate probability distribution based loss for unbiased scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19447–19456, June 2022.

[38] Yikang Li, Wanli Ouyang, Bolei Zhou, Kun Wang, and Xiaogang Wang. Scene graph generation from objects, phrases and region captions. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1261–1270, 2017.

[39] Yiming Li, Xiaoshan Yang, and Changsheng Xu. Dynamic scene graph generation via anticipatory pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13874–13883, June 2022.

[40] Yue Liao, Si Liu, Fei Wang, Yanjie Chen, Chen Qian, and Jiashi Feng. Ppdm: Parallel point detection and matching for real-time human-object interaction detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[41] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 936–944, 2017.

[42] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 740–755, Cham, 2014. Springer International Publishing.

[43] Xin Lin, Changxing Ding, Jinquan Zeng, and Dacheng Tao. Gps-net: Graph property sensing network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[44] Xin Lin, Changxing Ding, Yibing Zhan, Zijian Li, and Dacheng Tao. Hl-net: Heterophily learning network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19476–19485, June 2022.

[45] Xin Lin, Changxing Ding, Jing Zhang, Yibing Zhan, and Dacheng Tao. Ru-net: Regularized unrolling network for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19457–19466, June 2022.

[46] Bingbin Liu, Ehsan Adeli, Zhangjie Cao, Kuan-Hui Lee, Abhijeet Shenoi, Adrien Gaidon, and Juan Carlos Niebles. Spatiotemporal relationship reasoning for pedestrian intent prediction, 2020.

[47] Hengyue Liu, Ning Yan, Masood S. Mortazavi, and Bir Bhanu. Fully convolutional scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[48] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10012–10022, October 2021.

[49] Cewu Lu, Ranjay Krishna, Michael Bernstein, and Li Fei-Fei. Visual relationship detection with language priors. In *European Conference on Computer Vision*, 2016.

[50] Yichao Lu, Himanshu Rai, Jason Chang, Boris Knyazev, Guangwei Yu, Shashank Shekhar, Graham W. Taylor, and Maksims Volkovs. Context-aware scene graph generation with seq2seq transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15931–15941, October 2021.

[51] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the effective receptive field in deep convolutional neural networks. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[52] Xinyu Lyu, Lianli Gao, Yuyu Guo, Zhou Zhao, Hao Huang, Heng Tao Shen, and Jingkuan Song. Fine-grained predicates learning for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19467–19475, June 2022.

[53] Tim Meinhardt, Alexander Kirillov, Laura Leal-Taixé, and Christoph Feichtenhofer. Trackformer: Multi-object tracking with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8844–8854, June 2022.

[54] Taylor Mordan, Matthieu Cord, Patrick Pérez, and Alexandre Alahi. Detecting 32 pedestrian attributes for autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems (T-ITS)*, 2021.

[55] Alejandro Newell and Jia Deng. Pixels to graphs by associative embedding. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[56] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *European conference on computer vision*, pages 483–499. Springer, 2016.

[57] Bryan A Plummer, Arun Mallya, Christopher M Cervantes, Julia Hockenmaier, and Svetlana Lazebnik. Phrase localization and visual relationship detection with comprehensive image-language cues. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1928–1937, 2017.

[58] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12179–12188, 2021.

[59] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett, editors, *NIPS*, pages 91–99, 2015.

[60] Byungseok Roh, JaeWoong Shin, Wuhyun Shin, and Saehoon Kim. Sparse DETR: Efficient end-to-end object detection with learnable sparsity. In *International Conference on Learning Representations*, 2022.

[61] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[62] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE transactions on neural networks*, 20(1):61–80, 2008.

[63] Jing Shi, Yiwu Zhong, Ning Xu, Yin Li, and Chenliang Xu. A simple baseline for weakly-supervised scene graph generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16393–16402, October 2021.

[64] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *3rd International Conference on Learning Representations, ICLR 2015, Conference Track Proceedings*, 2015.

[65] Mohammed Suhail, Abhay Mittal, Behjat Siddiquie, Chris Broaddus, Jayan Eledath, Gerard Medioni, and Leonid Sigal. Energy-based learning for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13936–13945, June 2021.

[66] Zhiqing Sun, Shengcao Cao, Yiming Yang, and Kris M Kitani. Rethinking transformer-based set prediction for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3611–3620, 2021.

[67] Kaihua Tang, Yulei Niu, Jianqiang Huang, Jiaxin Shi, and Hanwang Zhang. Unbiased scene graph generation from biased training. In *Conference on Computer Vision and Pattern Recognition*, 2020.

[68] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. Learning to compose dynamic tree structures for visual contexts. In *Conference on Computer Vision and Pattern Recognition*, 2019.

[69] Kaihua Tang, Hanwang Zhang, Baoyuan Wu, Wenhan Luo, and Wei Liu. Learning to compose dynamic tree structures for visual contexts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[70] Yao Teng and Limin Wang. Structured sparse r-cnn for direct scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19437–19446, June 2022.

[71] Yao Teng, Limin Wang, Zhifeng Li, and Gangshan Wu. Target adaptive context aggregation for video scene graph generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 13688–13697, October 2021.

[72] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia

Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[73] Johanna Wald, Helisa Dhamo, Nassir Navab, and Federico Tombari. Learning 3d semantic scene graphs from 3d indoor reconstructions. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[74] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, et al. Deep high-resolution representation learning for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 2020.

[75] Tiancai Wang, Tong Yang, Martin Danelljan, Fahad Shahbaz Khan, Xiangyu Zhang, and Jian Sun. Learning human-object interaction detection using interaction points. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[76] Wenbin Wang, Ruiping Wang, and Xilin Chen. Topic scene graph generation by attention distillation from caption. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15900–15910, October 2021.

[77] Wenbin Wang, Ruiping Wang, Shiguang Shan, and Xilin Chen. Exploring context and visual pattern of relationship for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8188–8197, 2019.

[78] Sanghyun Woo, Dahun Kim, Donghyeon Cho, and In So Kweon. Linknet: Relational embedding for scene graph. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

[79] Shun-Cheng Wu, Johanna Wald, Keisuke Tateno, Nassir Navab, and Federico Tombari. Scenegraphfusion: Incremental 3d scene graph prediction from rgb-d sequences. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7515–7525, June 2021.

[80] Saining Xie, Ross Girshick, Piotr Dollar, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[81] Danfei Xu, Yuke Zhu, Christopher Choy, and Li Fei-Fei. Scene graph generation by iterative message passing. In *Computer Vision and Pattern Recognition (CVPR)*, 2017.

[82] Shaotian Yan, Chen Shen, Zhongming Jin, Jianqiang Huang, Rongxin Jiang, Yaowu Chen, and Xian-Sheng Hua. *PCPL: Predicate-Correlation Perception Learning for Unbiased Scene Graph Generation*, page 265–273. Association for Computing Machinery, New York, NY, USA, 2020.

[83] Gengcong Yang, Jingyi Zhang, Yong Zhang, Baoyuan Wu, and Yujiu Yang. Probabilistic modeling of semantic ambiguity for scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12527–12536, June 2021.

[84] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. Graph r-cnn for scene graph generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[85] Xu Yang, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. Auto-encoding scene graphs for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10685–10694, 2019.

[86] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. Exploring visual relationship for image captioning. In *Proceedings of the European conference on computer vision (ECCV)*, pages 684–699, 2018.

[87] Yuan Yao, Ao Zhang, Xu Han, Mengdi Li, Cornelius Weber, Zhiyuan Liu, Stefan Wermter, and Maosong Sun. Visual distant supervision for scene graph generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 15816–15826, October 2021.

[88] Keren Ye and Adriana Kovashka. Linguistic structures as weak supervision for visual scene graph generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8289–8299, June 2021.

[89] Ruichi Yu, Ang Li, Vlad I Morariu, and Larry S Davis. Visual relationship detection with internal and external linguistic knowledge distillation. In *Proceedings of the IEEE international conference on computer vision*, pages 1974–1982, 2017.

[90] Alireza Zareian, Svebor Karaman, and Shih-Fu Chang. Bridging knowledge graphs to generate scene graphs. In *European Conference on Computer Vision*, pages 606–623. Springer, 2020.

[91] Alireza Zareian, Zhecan Wang, Haoxuan You, and Shih-Fu Chang. Learning visual commonsense for robust scene graph generation. In *Computer Vision – ECCV 2020*, pages 642–657, 2020.

[92] Duncan Zauss, Sven Kreiss, and Alexandre Alahi. Keypoint communities. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11057–11066, 2021.

[93] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. Neural motifs: Scene graph parsing with global context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5831–5840, 2018.

[94] Chaoyi Zhang, Jianhui Yu, Yang Song, and Weidong Cai. Exploiting edge-oriented reasoning for 3d point-based scene graph analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9705–9715, June 2021.

[95] Hanwang Zhang, Zawlin Kyaw, Shih-Fu Chang, and Tat-Seng Chua. Visual translation embedding network for visual relation detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5532–5540, 2017.

[96] Ji Zhang, Kevin J Shih, Ahmed Elgammal, Andrew Tao, and Bryan Catanzaro. Graphical contrastive losses for scene graph parsing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11535–11543, 2019.

[97] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao

Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6881–6890, 2021.

[98] Yiwu Zhong, Jing Shi, Jianwei Yang, Chenliang Xu, and Yin Li. Learning to generate scene graph from natural language supervision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1823–1834, October 2021.

[99] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. In *arXiv preprint arXiv:1904.07850*, 2019.

[100] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable {detr}: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations*, 2021.

[101] Cheng Zou, Bohan Wang, Yue Hu, Junqi Liu, Qian Wu, Yu Zhao, Boxun Li, Chenguang Zhang, Chi Zhang, Yichen Wei, et al. End-to-end human object interaction detection with hoi transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11825–11834, 2021.