Reformulating HOI Detection as Adaptive Set Prediction

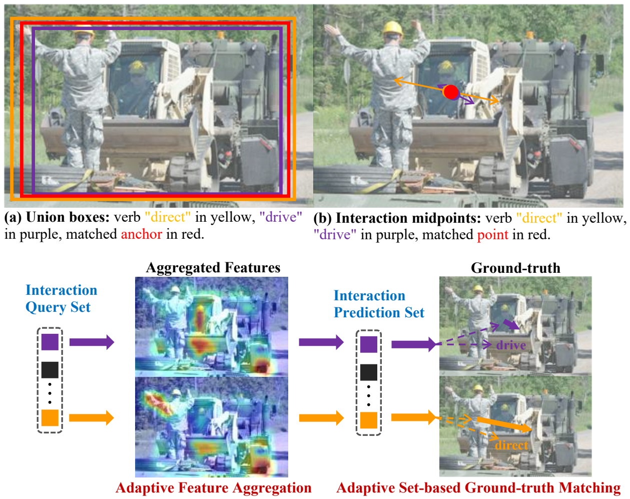
Mingfei Chen[1,3*] Yue Liao[2*] Si Liu[2†] Zhiyuan Chen[3] Fei Wang[3] Chen Qian[3]
[1] Huazhong University of Science and Technology
[2] Institute of Artificial Intelligence, Beihang University [3] SenseTime Research

Abstract

Determining which image regions to concentrate is critical for Human-Object Interaction (HOI) detection. Conventional HOI detectors focus on either detected human and object pairs or pre-defined interaction locations, which limits its learning of the effective features. In this paper, we reformulate HOI detection as an adaptive set prediction problem, with this novel formulation, we propose an Adaptive Set-based one-stage framework (AS-Net) with parallel instance and interaction branches. To attain this, we map a trainable interaction query set to an interaction prediction set with transformer. Each query adaptively aggregates the interaction-relevant features from global contexts through multi-head co-attention. Besides, the training process is supervised adaptively by matching each ground-truth with the interaction prediction. Furthermore, we design an effective instance-aware attention module to introduce instructive features from the instance branch into the interaction branch. Our method outperforms previous state-of-the-art methods without any extra human pose and language features on three challenging HOI detection datasets. Especially, we achieve over 31% *relative improvement on a large scale HICO-DET dataset. Code is available at* `https://github.com/yoyomimi/AS-Net`.

1. Introduction

Human-Object Interaction (HOI) detection aims to identify HOI triplets <human, verb, object> from a given image, it is an important step toward the high-level semantic understanding [8, 26, 46, 18, 17, 19, 6, 7, 44]. Conventional HOI methods can be divided into two-stage methods [38, 3, 10, 25, 24, 14, 9, 35] and one-stage methods [20, 27]. Most two-stage methods detect instances (humans and objects), and match the detected humans and objects one by one to form pair-wise proposals in the first stage. Next, in the second stage, such methods infer the interactions based on the features of cropped human-object pair-wise proposals. Two-stage methods have made great progress in HOI detection, however, their efficiency and effectiveness are limited by their serial architectures. With the development of one-stage object detectors, one-stage HOI detectors [20, 27] have raised a new fashion. Existing one-stage HOI detectors formulate HOI detection as a parallel detection problem, which detects the HOI triplets from an image directly. One-stage methods have delivered great improvements in both efficiency and effectiveness.


Aggregated Features
Ground-truth
Interaction Query Set
Interaction Prediction Set
drive
direct
Adaptive Feature Aggregation
Adaptive Set-based Ground-truth Matching

(a) Union boxes: verb "direct" in yellow, "drive" in purple, matched anchor in red. **(b) Interaction midpoints:** verb "direct" in yellow, "drive" in purple, matched point in red.

(c) Our adaptive set prediction method: verb "drive" in purple, "direct" in yellow. Interaction vectors point from human centers to object centers. The features aggregated by queries are visualized at left.

Figure 1. Both the anchor-based (a) and point-based (b) one-stage methods infer two different interactions "drive" and "direct" are at similar location and concentrate on the similar features. Our set prediction method (c) maps an interaction query set to an interaction prediction set by an interaction decoder. Then, interaction predictions are adaptively matched with ground-truth. To attain this, we first train a set of learnable embeddings as an interaction query set. Next, each interaction query adaptively aggregates the interaction-relevant features by co-attention. Finally, we match each ground-truth with prediction for adaptive supervision. This mechanism empowers our method to accurately predict two interactions for "drive" and "direct". Best viewed in color.

Determining which regions to concentrate on is critical and challenging for HOI detectors. To obtain essential features for interaction prediction, conventional two-

[*] Equal contribution
[†] Corresponding author (liusi@buaa.edu.cn)

stage methods usually involve extra features, *e.g.*, human pose [38, 5, 24, 14] and language [43, 9, 30, 21]. However, even with extra features, two-stage methods still focus on the detected instances that might be inaccurate, which are less adaptive and limited by the detected instances. One-stage methods partially alleviate these issues by inferring interactions directly from the whole image. Such methods intuitively define a location-relative medium to predict interactions, and can be mainly divided into anchor-based methods and point-based methods. Anchor-based methods [20] predict the interactions based on the union box of each pair-wise human and object instances. While point-based methods [27] infer the interaction midpoint of each corresponding human-object pair. However, we argue that it is sub-optimal to predict the interaction through a pre-defined interaction location. Figure 1 illustrates an example. The interaction "direct" (in yellow) and "drive" (in purple) are quite different and thus require different visual features for interaction prediction. However, their union boxes are considerably overlapped (Figure 1 (a)), and their interaction midpoints are very close (Figure 1 (b)). Therefore, these one-stage methods concentrate on similar visual features for the two different interactions.

To further address the limitation of interaction location in one-stage methods, we reformulate interaction detection as a set-based prediction problem. We define an interaction query set with several learnable embeddings, and an interaction prediction set. Each embedding in the query set is mapped by a transformer based interaction decoder to an interaction prediction set. By feeding the interaction query set into a multi-head co-attention module, we are able to adaptively aggregate features from global contexts. Our proposed method matches each ground-truth with the resembling interaction prediction for adaptive supervision. Therefore, our proposed method adaptively concentrates on the most suitable features for each prediction, free from the location limitation of conventional one-stage methods. As demonstrated in Figure 1 (c), our method aggregates arm features of the left person and pose features of the right person to make two different interaction predictions. The predictions are then matched with the ground-truth interaction "direct" and "drive" respectively.

To this end, we propose a novel Adaptive Set-based one-stage framework, namely AS-Net. Our AS-Net consists of two parallel branches: an instance branch and an interaction branch. Both branches leverage a transformer encoder-decoder structure, which utilize global features to perform set predictions. The instance branch predicts location and category for each instance, while the interaction branch predicts interaction vectors and their corresponding categories. The interaction vectors point from the centers of the human instances to the centers of the object instances. We obtain the predicted interaction triplets by matching each interaction vector from the interaction branch with the detected instances from the instance branch. Besides, we exploit an instance-aware attention module in a co-attention manner to perform branch aggregation. Specifically, this module aggregates information in the instance branch and introduces the aggregated features into the interaction branch. We also utilize semantic embeddings to perform more accurate human-object matching.

We test our proposed AS-Net on three datasets, *i.e.*, HICO-Det [34], V-COCO [13], and HOI-A [27]. Our proposed AS-Net outperforms all the other algorithms among all datasets. In specific, our proposed AS-Net has gained 31% relative improvement comparing to the previous state-of-the-art one-stage method [27] on HICO-DET.

Our contributions can be concluded in the following three aspects:

- We formulate HOI detection as a set prediction problem, which breaks the instance-centric limitation and location limitation of the existing methods. Thereby, our method can adaptively concentrate on the most suitable features to improve the predicting accuracy.
- We propose a novel one-stage transformer-based HOI detection framework, namely AS-Net. We also design an instance-aware attention module to introduce the information in the instance branch into the interaction branch.
- Without introducing any extra features, our method outperforms all the previous state-of-the-art methods, achieving 31% relative improvement over the second best one-stage method on the HICO-DET dataset.

2. Related Work

Two-stage Methods. Most conventional HOI detectors are in a two-stage manner. In the first stage, an object detector [11, 36, 4] is applied to detect the instances. In the second stage, the cropped instance features are classified to obtain the interaction categories. In addition to the cropped instance features, previous methods leverage combined spatial features [3, 12, 10, 14, 9, 16, 47], union box features [34, 40], or context features [10, 41, 30] to improve the accuracy of HOI detection. In order to concentrate on more interaction-relevant features, some methods utilize extra features, such as human pose [38, 5, 24, 14], human parts [48, 40, 23] and language features [43, 9, 30, 21]. However, the serial architectures of such two-stage methods impair the efficiency of HOI detection. Moreover, the prediction accuracy is usually limited by the results of instance detection.

One-stage Methods. Recently, one-stage HOI detection methods with higher efficiency [20, 27] have attracted increasing attention. Most one-stage methods extract features with a bottom-up structure[32, 45], and detect the

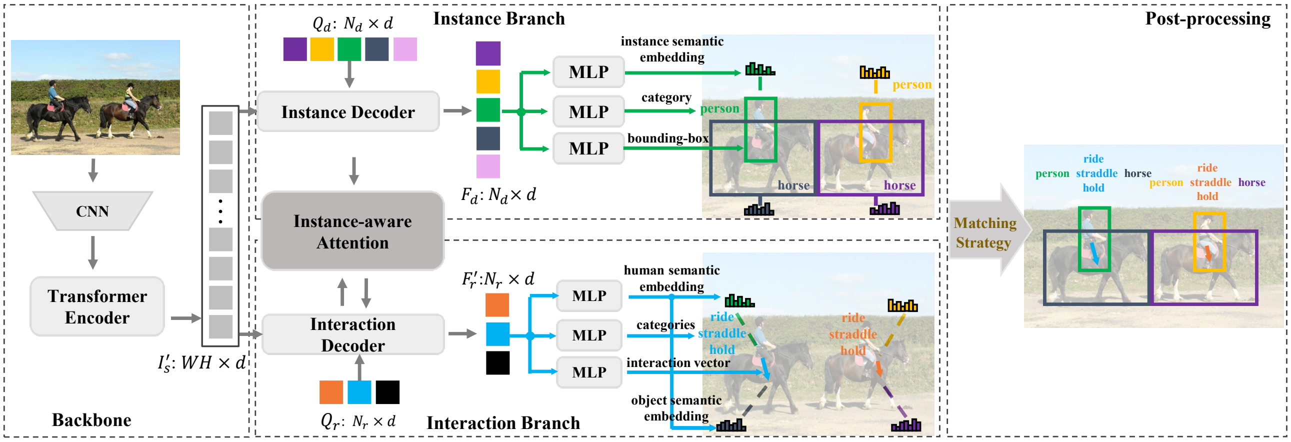


Figure 2. Overview of the proposed framework. First, a CNN and a transformer encoder are applied to extract the feature sequence with global contexts. Then two branches are built on the transformer decoder layers: a) the instance branch transforms a set of learnable instance queries to an instance prediction set one by one b) the interaction branch utilizes an interaction query set to estimate an interaction prediction set. The instance-aware attention module is designed to introduce the interaction-relevant instance features from the instance branch to the interaction branch. At the end, the detected instances are matched with the interaction predictions to infer the HOI triplets.

HOI triplets in parallel from an image directly. Specifically, the one-stage methods can be divided into anchor-based methods [20] and point-based methods [27] according to the manners of their interaction prediction. The anchor-based methods predict the interactions based on each union box. The point-based methods perform inference at each interaction key point, such as the midpoint of each corresponding human-object pair. Though breaking the limitation from instance detection, such methods which pre-assign each ground-truth interaction to the predictions, are still non-adaptive and limited by the interaction locations.

3. Methods

HOI detection aims to predict the triplet of <human, verb, object>, which contains a pair of bounding-boxes for a human and an object, and a corresponding verb category. In this paper, we reformulate HOI detection as a set prediction problem, and propose an Adaptive Set-based one-stage Network (AS-Net).

Our AS-Net builds on a transformer encoder-decoder architecture and makes parallel set-based predictions for the HOI triplets. As illustrated in Figure 2, our proposed AS-Net consists of four parts. We first utilize a backbone (Section 3.1) to extract the visual feature sequence with global contexts. The instance (Section 3.2) and interaction branches (Section 3.3) following the backbone parallelly detect an instance and interaction prediction set from the feature sequence respectively. In order to intensify the instance features that are valuable for interaction inference, we design an instance-aware attention module (Section 3.4) to perform branch aggregation. Specifically, we introduce semantic embeddings (Section 3.5) in instance and interaction branches for more accurate triplet prediction. At the end, we match detected instances and interactions to obtain the final HOI triplets (Section 3.6).

3.1. Backbone

We define the backbone by combining a CNN and a transformer encoder to extract the image features. The encoder is in a multi-layer manner, where each layer comprises a multi-head self-attention module and a two-layer Feed-Forward Network (FFN). For a given image, we first extract a visual feature map $\boldsymbol{I} \in \mathbb{R}^{W \times H \times C}$ using the CNN. Then we utilize a 1×1 convolution to reduce the channel dimension of the visual feature map from C to d, and reshape such feature map as a feature sequence $\boldsymbol{I}_s \in \mathbb{R}^{WH \times d}$. Next, we feed the feature sequence to the encoder which refines the feature sequence by introducing global contexts into the output feature sequence $\boldsymbol{I}'_s \in \mathbb{R}^{WH \times d}$.

3.2. Instance Branch

The instance branch is designed to localize and classify the instances. Following the detector DETR [2], our instance branch consists of a multi-layer transformer decoder and several FFN heads. Each layer of the decoder is comprised of a self-attention module, and a multi-head co-attention module. The input of each decoder layer is the summation of a learnable positional embedding sequence $\boldsymbol{Q}_d \in \mathbb{R}^{N_d \times d}$ and the output of last layer. Except for the first layer where there is no output of last layer, we added zeros to the learnable positional embedding sequence. We first feed the input into the self-attention module. Then the multi-head co-attention module adaptively aggregates the key contents from $\boldsymbol{I}'_s$ to $\boldsymbol{F}_d \in \mathbb{R}^{N_d \times d}$, where we take $\boldsymbol{Q}_d$ with the output of self-attention as queries, and $\boldsymbol{I}'_s$ with the corresponding fixed positional encodings [33] as keys. There is an FFN head on top of each decoder layer to decode a set of instance predictions from $\boldsymbol{F}_d$. The FFN head comprises three independent sub-branches. One to predict the normalized bounding-box in (cx, cy, w, h) format for each detected instance. Another to infer a $(L_d + 1)$-dimensional scores for L_d categories, where the last dimension refers to the no-object ($\varnothing$) category. The other to generate a distinctive semantic embedding $\boldsymbol{\varepsilon} \in \mathbb{R}^K$ for each instance, which will be explained in Section 3.5. Each sub-branch constitutes of one or several perception layers. The FFN head of each decoder layer shares the same weights.

Training. For the set-based training process, we first find a one-to-one bipartite matching between the detected instance set $\hat{y}$ and the ground-truth y (padded with no-instance $\varnothing$ to a set of size N_d). To this end, we deploy a matching loss, which is the summation of bounding-box loss and category semantic distance between instance and all ground-truth bounding-boxes. Following [2], the bounding-box loss is composed of a l_1 loss and a GIoU loss [37]. The category semantic distance is the negative of summation of the predicted scores for each ground-truth category.

The universal index permutation set of N_d predictions is denoted as S_{N_d}. We consider $\hat{\sigma}_d \in S_{N_d}$ that minimizes the summation of all the matching cost $\mathcal{L}_{\text{match}}(\hat{y}_{\sigma_d(i)}, y_i)$ as the optimal index permutation of the detected instance set, which we adopt the Hungarian algorithm [22] to calculate. The i-th element of the index permutation $\sigma_d \in S_{N_d}$ is defined as $\sigma_d(i)$, and the $\hat{\sigma}_d$ is formulated as:

$$\hat{\sigma}_d = \underset{\sigma_d \in S_{N_d}}{\arg\min} \sum_{i=1}^{N_d} \mathbb{1}_{y_i \neq \varnothing} \mathcal{L}_{\text{match}}(\hat{y}_{\sigma_d(i)}, y_i). \tag{1}$$

For the instance prediction with index permutation $\hat{\sigma}_d(i)$, the predicted bounding-box and category are represented as $\hat{b}_{\hat{\sigma}_d(i)}$ and $\hat{p}_{\hat{\sigma}_d(i)}$ respectively. We follow the DETR detector [2] to construct the set-based instance detection loss $\mathcal{L}_{\text{ins}}$:

$$\mathcal{L}_{\text{ins}} = \sum_{i=1}^{N_d} [-\log \hat{p}_{\hat{\sigma}_d(i)}(c_i) + \mathbb{1}_{\{c_i \neq \varnothing\}} \mathcal{L}_{\text{box}}(b_i, \hat{b}_{\hat{\sigma}_d(i)})], \tag{2}$$

where b_i and c_i denotes the bounding-box and category of the matched ground-truth instance respectively, $\hat{p}_{\hat{\sigma}_d(i)}(c_i)$ is the confidence score for category c_i.

3.3. Interaction Branch

The interaction branch predicts the interaction vectors and categories for each interaction. Its architecture is similar to the instance branch, which constitutes a multi-layer transformer decoder and several FFN heads. Each decoder layer utilizes several interaction query set $\boldsymbol{Q}_r$ to aggregate the corresponding key contents $\boldsymbol{F}_r \in \mathbb{R}^{N_r \times d}$ from the shared feature sequence $\boldsymbol{I}'_s$. Each decoder layer is equipped with a FFN head as the instance branch. Each FFN head is also split into three sub-branches. For each interaction prediction, we predict a 4-dimensional interaction vector with categories, and two semantic embeddings, *i.e.*, $\boldsymbol{\varepsilon}^h \in \mathbb{R}^K$ and $\boldsymbol{\varepsilon}^o \in \mathbb{R}^K$ for the corresponding human and object instances respectively. The interaction vector points from the normalized human center (x^h_{ct}, y^h_{ct}) to the object center (x^o_{ct}, y^o_{ct}). Considering there might exist multiple interactions for the same human-object pair, we use a multi-label classifier to predict a score for each verb category respectively.

Training. We denote the ground-truth interaction as $t = (v, z)$, where v is the interaction vector of t, and z indicates the L ground-truth interaction categories of t. We compute the matching loss between t and each predicted interaction $\hat{t} = (\hat{v}, \hat{z})$, where $\hat{v}$ refers to the predicted interaction vector and $\hat{z}$ indicates the confidence scores of the interaction categories. The matching cost $\mathcal{L}_{\text{match}}(\hat{t}_{\sigma_r(i)}, t_i)$ can be computed by:

$$\mathcal{L}_{\text{match}}(\hat{t}_{\sigma_r(i)}, t_i) = \|v_i - \hat{v}_{\sigma_r(i)}\|_1 + \sum_{l=1}^{L} - \frac{1}{1 + e^{-\hat{z}_{\sigma_r(i)}(z_l)}}, \tag{3}$$

where $\hat{z}_{\sigma_r(i)}(z_l)$ refers to the score for the l-th ground-truth interaction category z_l of t. Similar to the set-based training process for the instance branch, we utilize the Hungarian algorithm [22] to find the optimal index assignment $\hat{\sigma}_r$ for the predicted interaction set *w.r.t.* the ground-truth.

For the interaction prediction with index $\hat{\sigma}_r(i)$, we define the predicted interaction vector and categories as $\hat{v}_{\hat{\sigma}_r(i)}$ and $\hat{z}_{\hat{\sigma}_r(i)}$ respectively, and the matched target interaction vector and categories are v_i and z_i respectively. To balance the ratio between the positive and negative samples for each classifier, we apply Focal loss [28], denoted as $\mathcal{L}_{\text{cls}}$, for the training of interaction classification. Besides, we adopt l_1 loss, denoted as $\mathcal{L}_{\text{reg}}$, for the regression of interaction vectors. The interaction loss $\mathcal{L}_{\text{int}}$ is calculated as:

$$\begin{aligned} \mathcal{L}_{\text{int}} = \sum_{i=1}^{N_r} & [\lambda_{\text{cls}} \mathcal{L}_{\text{cls}}(z_i, \hat{z}_{\hat{\sigma}_r(i)}) \\ & + \mathbb{1}_{\{z_i \neq \varnothing\}} \lambda_{\text{reg}} \mathcal{L}_{\text{reg}}(v_i, \hat{v}_{\hat{\sigma}_r(i)})], \end{aligned} \tag{4}$$

where λ_{cls} and λ_{reg} are the weight coefficients of $\mathcal{L}_{\text{cls}}$ and $\mathcal{L}_{\text{reg}}$ respectively.

Analysis. Adaptation is involved in the interaction prediction from two aspects. First, for each interaction query, we apply multi-head co-attention to aggregate information from each element in the feature sequence. Hence, each query can adaptively aggregate the interaction-relevant visual features. Second, instead of pre-assigning each ground-truth to the corresponding prediction, we consider both the predicted interaction vectors and categories to match each ground-truth interaction with the resembling prediction. Therefore, each interaction prediction can be supervised by the most suitable ground-truth more adaptively.

3.4. Instance-aware Attention

We construct an instance-aware attention module between each instance and interaction layer to emphasize relevant instance features for interaction prediction.

First, we compute an affinity score map $\boldsymbol{A} \in \mathbb{R}^{(N_r \times N_d)}$ between the instance features $\boldsymbol{F}_d$ and the interaction features $\boldsymbol{F}_r$:

$$\boldsymbol{A} = \frac{(\boldsymbol{W}_r\boldsymbol{F}_r + b_r)(\boldsymbol{W}_d\boldsymbol{F}_d + b_d)^\top}{\sqrt{d}}. \tag{5}$$

We then apply Softmax to obtain the instance-aware attention weight matrix $\boldsymbol{M} \in [0,1]^{(N_r \times N_d)}$:

$$\boldsymbol{M}_{ij} = \frac{exp(\boldsymbol{A}_{ij})}{\sum_{j=1}^{N_d} exp(\boldsymbol{A}_{ij})}, \tag{6}$$

where $\boldsymbol{M}_{ij}$ refers to the attention weight of the j-th detected instance with respect to the i-th predicted interaction. The final output interaction features $\boldsymbol{F}_r' \in \mathbb{R}^{(N_r \times d)}$ of the instance-aware attention module is formulated as:

$$\boldsymbol{F}_r' = \boldsymbol{M}(\boldsymbol{W}_d'\boldsymbol{F}_d + b_d') + \boldsymbol{F}_r. \tag{7}$$

3.5. Semantic Embedding

The interaction vectors are not pointing to a instance directly, instead, they point to a region. Instead of matching which only employs location indication from the interaction vectors, we introduce semantic embeddings inferred by an MLP block in our matching strategy. We infer semantic embeddings $\boldsymbol{\varepsilon}$ from $\boldsymbol{F}_d$ for each detected instance in instance branch. And in the interaction branch, two semantic embeddings $\boldsymbol{\varepsilon}^h$ and $\boldsymbol{\varepsilon}^o$ are inferred from $\boldsymbol{F}_r'$, one for the human instance and another for the object instance for each prediction.

In the training process, the semantic embeddings of different instances are pushed away from each other. The push procedure can be described as:

$$\mathcal{L}_{\text{push}} = \sum_{i=1}^{|\hat{\sigma}_d|-1} \sum_{j=i+1}^{|\hat{\sigma}_d|} [\max(0, t - \|\boldsymbol{\varepsilon}_{\hat{\sigma}_d(i)} - \boldsymbol{\varepsilon}_{\hat{\sigma}_d(j)}\|)]^2, \tag{8}$$

where $|\hat{\sigma}_d|$ refers to the total number of the ground-truth instances, and $\boldsymbol{\varepsilon}_{\hat{\sigma}_d(i)}$ refers to the semantic embedding of the predicted instance matched to the i-th target instance. If the l_2 distance between two semantic embeddings are more than a threshold t, we consider two embeddings are separate enough and set $\mathcal{L}_{\text{push}}$ to 0.

We pull the semantic embeddings that refer to the same instance towards each other:

$$\mathcal{L}_{\text{pull}} = \sum_{i=1}^{|\hat{\sigma}_r|} (\|\boldsymbol{\varepsilon}^h_{\hat{\sigma}_r(i)} - \boldsymbol{\varepsilon}_{\hat{\sigma}_d(h_i)}\|^2 + \|\boldsymbol{\varepsilon}^o_{\hat{\sigma}_r(i)} - \boldsymbol{\varepsilon}_{\hat{\sigma}_d(o_i)}\|^2), \tag{9}$$

where we denote the predicted human semantic embedding as $\boldsymbol{\varepsilon}^h_{\hat{\sigma}_r(i)}$ and the object embedding as $\boldsymbol{\varepsilon}^o_{\hat{\sigma}_r(i)}$ for the interaction prediction with index $\hat{\sigma}_r(i)$. The semantic embedding $\boldsymbol{\varepsilon}_{\hat{\sigma}_d(h_i)}$ and $\boldsymbol{\varepsilon}^h_{\hat{\sigma}_r(i)}$ refer to the same human instance in the instance and interaction branches respectively. Similarly, $\boldsymbol{\varepsilon}_{\hat{\sigma}_d(o_i)}$ and $\boldsymbol{\varepsilon}^o_{\hat{\sigma}_r(i)}$ refer to the same object instance. $|\hat{\sigma}_r|$ refers to the total number of the ground-truth interactions.

3.6. Training Loss and Post-processing

The target loss is the weighted sum of the losses mentioned above:

$$\mathcal{L} = \mathcal{L}_{\text{ins}} + \mathcal{L}_{\text{int}} + \lambda_{\text{emb}}(\mathcal{L}_{\text{pull}} + \mathcal{L}_{\text{push}}), \tag{10}$$

where λ_{emb} is a hyper-parameter to balance different loss.

During the post-processing, we first match the detected human instances with object instances based on our predicted interaction vectors and semantic embeddings. A good human-object interaction match should meet the following three requirements: 1) the normalized center of the matched human/object instances is close to the start and the end point of the interaction vector respectively; 2) the matched instances have high confidence scores on their predicted categories; 3) the semantic embedding referring to the same matched instances are similar to each other.

We consider all detected instances as object instances. For each predicted interaction vector $\hat{v} = (\hat{x}^h_{ct}, \hat{y}^h_{ct}, \hat{x}^o_{ct}, \hat{y}^o_{ct})$, the matching distance $\mathcal{D}$ can be calculated as:

$$\begin{aligned} \mathcal{D} =& (|\tilde{x}^h_{ct} - \hat{x}^h_{ct}| + 1)(|\tilde{y}^h_{ct} - \hat{y}^h_{ct}| + 1) \\ & (|\tilde{x}^o_{ct} - \hat{x}^o_{ct}| + 1)(|\tilde{y}^o_{ct} - \hat{y}^o_{ct}| + 1), \end{aligned} \tag{11}$$

where $(\tilde{x}^h_{ct}, \tilde{y}^h_{ct})$, $(\tilde{x}^o_{ct}, \tilde{y}^o_{ct})$ refers to the center of the detected human and object instances, with a confidence score of s^h and s^o, respectively.

When the semantic embeddings are introduced for matching, given the human ($\boldsymbol{\varepsilon}_h$) and object ($\boldsymbol{\varepsilon}_o$) semantic embedding of the instance branch, the embedding matching distance $\mathcal{R}$ for the predicted human ($\hat{\boldsymbol{\varepsilon}}_h$) and object ($\hat{\boldsymbol{\varepsilon}}_o$) semantic embedding of the interaction branch can be defined as:

$$\mathcal{R} = (\|\boldsymbol{\varepsilon}_h - \hat{\boldsymbol{\varepsilon}}_h\| + 1)(\|\boldsymbol{\varepsilon}_o - \hat{\boldsymbol{\varepsilon}}_o\| + 1). \tag{12}$$

The final matching cost is calculated as $\frac{\mathcal{DR}}{s^h s^o}$. We match the detected instances with the minimum matching cost to each interaction prediction. The HOI confidence score for each predicted triplet is the product of the interaction category score, and the matched instance scores s^h and s^o. Triplets with top N confidence scores are preserved as the final HOI triplet predictions.

4. Experiments

4.1. Datasets and Metrics

Datasets. To verify the effectiveness of our model, we conduct experiments on three HOI detection datasets HICO-DET [3], V-COCO [13] and HOI-A [27]. HICO-DET contains $38,118$ images for training and $9,658$ images for testing, contains the same 80 object categories as MS-COCO [29] and 117 verb categories. The objects and verbs form 600 classes of HOI triplets. V-COCO provides $2,533$ images for training, $2,867$ images for validating and $4,946$ images for testing. V-COCO is derived from MS-COCO dataset, annotated with 29 action categories. HOI-A dataset

Method	Backbone	Finetune Detection	Extra	Time (ms) / FPS	Default Full	Default Rare	Default Non-Rare	Know Object Full	Know Object Rare	Know Object Non-Rare
Two-stage Method:										
InteractNet [12]	ResNet-50-FPN	✗	✗	145 / 6.90	9.94	7.16	10.77	-	-	-
GPNN [34]	Res-DCN-152	✗	✗	-	13.11	9.34	14.23	-	-	-
iCAN [10]	ResNet-50	✗	✗	204 / 4.90	14.84	10.45	16.15	16.26	11.33	17.73
No-Frills [14]	ResNet-152	✗	*P*	494 / 2.02	17.18	12.17	18.68	-	-	-
PMFNet [40]	ResNet-50-FPN	✗	*P*	253 / 3.95	17.46	15.65	18.00	20.34	17.47	21.20
DRG [9]	ResNet-50-FPN	✗	*L*	200 / 5.00	19.26	17.74	19.71	23.40	21.75	23.89
IP-Net [42]	Hourglass-104	✗	✗	-	19.56	12.79	21.58	22.05	15.77	23.92
VSGNet [39]	ResNet-152	✗	✗	312 / 3.21	19.80	16.05	20.91	-	-	-
PD-Net [47]	ResNet-152-FPN	✗	*L*	-	20.81	15.90	22.28	24.78	18.88	26.54
DJ-RN [23]	ResNet-50	✗	*P*	-	21.34	18.53	22.18	23.69	20.64	24.60
One-stage Method:										
UnionDet [20]	ResNet-50-FPN	✓	✗	78 / 12.82	17.58	11.72	19.33	19.76	14.68	21.27
PPDM-Hourglass [27]	Hourglass-104	✓	✗	71 / 14.08	21.94	13.97	24.32	24.81	17.09	27.12
AS-Net*	ResNet-50	✗	✗	71 / 14.08	24.40	22.39	25.01	27.41	25.44	28.00
AS-Net	ResNet-50	✓	✗	**71 / 14.08**	**28.87**	**24.25**	**30.25**	**31.74**	**27.07**	**33.14**

Table 1. **Performance comparison on the HICO-DET test set.** The 'P', 'L' represent human pose information and the language feature, respectively. * denotes freezing the instance detection related parameters pretrained on the MS-COCO dataset. Our one-stage model with a high inference speed of 71 ms / 14.08 FPS outperforms all previous work by a large margin.

consists of $38,668$ annotated images, 11 kinds of objects and 10 action categories.

Metrics. Following [3], the mean average precision (mAP) is adopted as evaluation metric. For one positive predicted HOI triplet $\langle$human, verb, object$\rangle$, both the predicted human and object bounding-boxes have IoUs greater than 0.5 *w.r.t.* the ground-truth boxes, with the correct predicted verb simultaneously.

Method	Backbone	Extra	mAP$_{role}$
Two-stage Method:			
InteractNet [12]	ResNet-50-FPN	✗	40.0
GPNN *et al.* [34]	Res-DCN-152	✗	44.0
iCAN [10]	ResNet-50	✗	45.3
DRG [9]	ResNet-50-FPN	*L*	51.0
IP-Net [42]	Hourglass-104	✗	51.0
VSGNet [39]	ResNet-152	✗	51.8
PMFNet [40]	ResNet-50-FPN	*P*	52.0
PD-Net [47]	ResNet-152-FPN	*L*	52.6
FCMNet [30]	ResNet-50	✗	53.1
One-stage Method:			
UnionDet [20]	ResNet-50-FPN	✗	47.5
AS-Net*	ResNet-50	✗	**53.9**

Table 2. **Performance comparison on the V-COCO test set**. The 'P', 'L' represent the human pose information and the language feature, respectively. * denotes freezing the instance detection related parameters pretrained on the MS-COCO dataset.

4.2. Implementation Details

Our implementation is based on two parallel 6-layer transformer decoders with a shared backbone, where the backbone is built on ResNet-50 [15] with a 6-layer self-attention encoder. Following the detector DETR [2], we infer $N_d = 100$ instances based on the aggregated interaction-relevant contents $\boldsymbol{F}_d \in \mathbb{R}^{100\times 256}$ on the instance branch.

Method	Backbone	Extra	mAP
Two-stage Method:			
iCAN [10]	ResNet-50	✗	44.23
TIN [25]	ResNet-50	*P*	48.64
Faster Interaction Net [1]	ResNet-50	✗	56.93
GMVM [1]	ResNet-50	*P*	60.26
C-HOI [49]	ResNet-50	*P*	66.04
One-stage Method:			
PPDM-DLA [27]	DLA-34	✗	67.03
PPDM-Hourglass [27]	Hourglass-104	✗	71.23
AS-Net	ResNet-50	✗	**72.19**

Table 3. **Performance comparison on the HOI-A test set**. The 'P' represents the extra human pose or body parts information.

On the interaction branch, we infer a set of $N_r = 16$ interaction vectors with categories from $\boldsymbol{F}_r' \in \mathbb{R}^{16\times 256}$. Moreover, the predicted semantic embeddings are all with a dimension of $K = 8$. After the matching process 3.6, the top $N = 100$ predictions are finally preserved.

During training, we resize the shortest side of the input image to the range $[480, 800]$, and the longest side is no more than $1,333$. We set the weight coefficients λ_{cls}, λ_{reg} and λ_{emb} in Section 3.6 to 1, 2 and 0.1 respectively. The model is trained with AdamW [31] for 90 epochs on the HICO-DET and HOI-A dataset, and for 75 epochs on the V-COCO dataset, with a learning rate of 10^{-4} decreased by 10 times at the 70th epoch. All the instance detection related parameters (backbone and the instance decoder layers) which are pretrained on the MS-COCO dataset, are frozen on the V-COCO dataset and trained with a learning rate of 10^{-5} on the other two datasets. Our experiments are all conducted on the GeForce GTX 1080Ti GPU and CUDA 9.0, with a batchsize of 64 on 32 GPUs.

Strategy	Full	Rare	Non-Rare
Vector	28.56	24.13	29.88
Embedding	28.65	23.95	30.05
Combined	**28.87**	**24.25**	**30.25**

(a) **Matching Strategy:** Analysis of different matching strategies, *i.e.*, interaction vector and semantic embeddings.

K	Full	Rare	Non-Rare	#Parameters
4	28.21	22.65	29.87	52.527 M
8	**28.87**	**24.25**	**30.25**	52.530 M
16	28.36	23.08	29.93	52.537 M
32	28.70	23.83	30.16	52.549 M

(b) **Dimension of Semantic Embeddings:** Choice of dimension of semantic embeddings.

λ_{emb}	Full	Rare	Non-Rare
0.05	28.31	23.65	29.70
0.1	**28.87**	**24.25**	**30.25**
0.5	27.84	21.71	29.67

(c) **Weight Coefficient λ_{emb}:** The effects of different settings of loss weight.

	Decoder Layers	Embeddings	IA Attention	Full	Rare	Non-Rare	#Parameters
Single Branch	6×	✗	-	25.91	17.88	28.31	41.44 M
Basic Model, Int×6	6×	✗	-	27.52	22.04	29.16	50.94 M
+ *IA Attn×6, Int w/o emb×6*	6×	✗	6×	27.96	23.01	29.44	52.13 M
+ *Int w/ emb×6*	6×	✓	-	27.75	22.71	29.25	51.34 M
+ *IA Attn×3, Int w/ emb×6*	6×	✓	3×	28.39	24.02	29.70	51.94 M
+ *IA Attn×3, Int w/ emb×3*	3×	✓	3×	28.63	23.61	30.13	47.20 M
+ *IA Attn×6, Int w/ emb×6*	6×	✓	6×	**28.87**	**24.25**	**30.25**	52.53 M

(d) **Component Analysis:** Results of the variants with various components, *i.e.*, interaction branches (Int), instance-aware attention module (IA Attn) and semantic embeddings (emb).

Table 4. Ablation studies of our proposed model on the HICO-DET test set.

4.3. Comparing to State-of-the-art

We conduct experiments on three HOI detection benchmarks to verify the effectiveness of our AS-Net. It is shown in Table 1, Table 2 and Table 3 that our AS-Net has achieved state-of-the-art across all the three benchmarks. Specifically, on the HICO-DET dataset, comparing to the previous state-of-the-art one-stage method PPDM [27] which adopts Hourglass-104 as backbone, our AS-Net has achieved a 31% performance gain with a relatively light-weight backbone, *i.e.*, ResNet-50. Since the object detectors in two-stage methods are purely trained on MS-COCO, which does not fine-tune on HICO-DET, thus we also show the result when only training the interaction branch for fair comparison. In this setting, our AS-Net* has achieved 24.40% mAP, which is superior to all existing two-stage methods, and has achieved above 3% mAP improvements.

We compare our results on the V-COCO dataset with other state-of-the-art methods. Freezing the instance detection related parameters pretrained on the MS-COCO dataset, we only train the remaining parameters of our model. As shown in Table 2, our model achieves 53.9% on mAP_{role}, outperforms the previous works. Considering the relatively small scale of the V-COCO dataset may impair the representation capability of the trained semantic embeddings, we test the results using the matching strategy without the semantic embeddings.

The Table 3 also illustrates our effectiveness on the HOI-A test set. We reach a mAP of 72.19%, better than all the previous methods, including the method which adopts a relatively heavy-weight Hourglass-104 as backbone.

4.4. Ablation Study

Matching Strategy. Two variants of inference matching methods are implemented. As shown in Table 4a, when only using the vector matching distance $\mathcal{D}$, or only using the semantic embedding distance $\mathcal{R}$ in Section 3.6, the effectiveness are both compromised.

Semantic Embedding Settings. To explore the suitable semantic embedding setting, we evaluate the models with different embedding dimension K and weight coefficient λ_{emb} of the training losses $\mathcal{L}_{\text{pull}}$ and $\mathcal{L}_{\text{push}}$. As shown in Table 4b, the effectiveness of our model is not sensitive to the embedding dimension. As K changes from 4 to 32, the changing of the mAP result is only 0.66 point. The embedding dimension K is set to 8 regarding the trade-off for both effectiveness and computational cost. As illustrated in Table 4c, the model performs best when training with $\lambda_{\text{emb}} = 0.1$, while the effectiveness will be impaired when λ_{emb} is increased or decreased.

Single Branch Variant. We implemented a single branch variant to detect instances along their interactions while keeping all hyper-parameters. As shown in Table 4d, the variant achieves 25.91% mAP on the HICO-DET dataset, which is 2.96% lower than our AS-Net. Especially, the Rare mAP is 17.88%, which is 6.37% lower than ours. We consider it is because detection and interaction rely on some *different features*. Lacking the interaction-related features such as human postures, the single branch variant is more likely to infer actions that frequently appear in the presence of the detected objects.

Basic Model. To verify the effectiveness of the basic framework, we implement a variant consists of one 6-layer instance detection branch and one 6-layer interaction detection branch, without the instance-aware interaction attention module and semantic embedding. Table 4d articulates that our basic model (Basic Model, Int×6) achieves 27.52% mAP on the HICO-DET dataset, which outperforms the previous methods by a large margin.

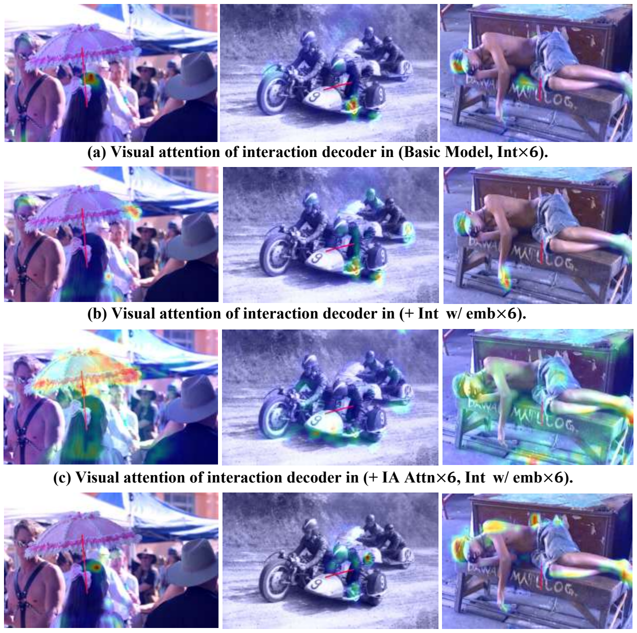
Instance-aware Attention. Two other variants are evaluated by utilizing the instance-aware attention module to verify the contribution of branch aggregation. As presented in Table 4d, the instance-aware attention module on our basic model (+ IA Attn×6, Int w/o emb×6) improves mAP by 0.44 point. For the basic model with the semantic embeddings (+ Int w/ emb×6), the improvements are 1.12 points using the instance-aware attention module (+ IA Attn×6, Int w/ emb×6). Therefore, we conclude that the instance-aware attention features from the instance branch are valuable for the interaction prediction.

Semantic Embedding & Instance-aware Attention. The basic model with the semantic embeddings (+ Int×6 w/ emb) improves slightly comparing to the basic model (Basic Model, Int×6) without the instance-aware attention as shown in Table 4d. As the bridge connecting predicted instances and interaction vectors, the semantic embeddings also contribute to the training. However, the semantic embedding is less powerful than the instance-aware interaction attention module from the results. Based on the basic model with the semantic embedding, several variants are implemented, which consist of different interaction decoder layers or attention modules additionally:1) 3 instance-aware attention modules with 6-layer interaction decoder (+ IA Attn×3, Int w/ emb×6), performs attention every other layer; 2) 3 instance-aware attention modules with 3-layer interaction decoder (+ IA Attn×3, Int w/ emb×3). From Table 4d, the performance is improved by about 1 point utilizing the attention module and the semantic embedding jointly. Besides, it's better to use the instance-aware modules and the decoder layers with the same number of times. The effectiveness reduces slightly when we utilize the two modules with less times, while the amount of the model parameters is reduced significantly.

4.5. Qualitative Results

As shown in the first three rows of the Figure 3, we visualize the interaction decoder attention for some interaction pairs in our basic model, the basic model with the semantic embeddings (+ Int w/ emb×6) and the model with both the instance-aware attention module as well as the semantic embeddings (+ IA Attn×6, Int w/ emb×6), respectively. We also visualize the instance-aware attention in the last row for each example interaction pairs to present how the attention module contributes to the interaction prediction.

From the Figure 3 (a), the basic model without any branch aggregation focuses on some scattered redundant feature regions and leave out some interaction-relevant features. From the Figure 3 (b), the model with semantic embeddings only partially alleviates the problem. For example, there is a girl holding an umbrella in the figures in the first column. To predict such interaction, the basic model concentrates on the head of the girl and the body of an irrelevant person. Correspondingly, the model with semantic embeddings pays attention to the edge of the umbrella and the body of the girl, while still concentrates on an irrelevant person. When we involve the instance-aware attention module, as shown in Figure 3 (c) and Figure 3 (d), the interaction branch concentrates on the whole umbrella and some body parts which are close to the umbrella, and the instance-aware attention module focuses on the body and head of the girl. In such a separated focus mechanism, our model can concentrate on the features more accurately.



(a) Visual attention of interaction decoder in (Basic Model, Int×6).

(b) Visual attention of interaction decoder in (+ Int w/ emb×6).

(c) Visual attention of interaction decoder in (+ IA Attn×6, Int w/ emb×6).

(d) Visual attention of instance-aware attention in (+ IA Attn×6, Int w/ emb×6).

Figure 3. Visualization of the interaction-relevant attention. In each sub-figure, the interaction vector in red is pointing from the corresponding human center to the object center.

5. Conclusion and Future Work

In this paper, we reformulate HOI detection as an adaptive set prediction problem and propose a novel one-stage HOI detection framework, namely AS-Net. By aggregating interaction-relevant features from global contexts, and matching each ground-truth with the interaction prediction, our method demonstrates adaptive ability on both feature aggregation and supervision. Moreover, the designed instance-aware attention module contributes to intensify the instructive instance features, and we also introduce semantic embeddings to improve performance. The ablation studies verify the effectiveness of each key component of our model. Our AS-Net outperforms all existing methods on three HOI detection datasets. In the future, we plan to extend AS-Net to handle more general association problems, *e.g.*, visual relationship detection and multi-object tracking.

Acknowledgement This work was partially supported by Sensetime Ltd. Group, National Natural Science Foundation of China under Grant 61876177, Zhejiang Lab (No. 2019KD0AB04), Beijing Natural Science Foundation 4202034 and Fundamental Research Funds for the Central Universities.

References

[1] Pic leaderboard. http://www.picdataset.com/challenge/leaderboard/hoi2019, 2019. 6

[2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 3, 4, 6

[3] Yu-Wei Chao, Yunfan Liu, Xieyang Liu, Huayi Zeng, and Jia Deng. Learning to detect human-object interactions. In *WACV*, 2018. 1, 2, 5, 6

[4] Zhiwei Dong, Guoxuan Li, Yue Liao, Fei Wang, Pengju Ren, and Chen Qian. Centripetalnet: Pursuing high-quality keypoint pairs for object detection. In *CVPR*, 2020. 2

[5] Wei Feng, Wentao Liu, Tong Li, Jing Peng, Chen Qian, and Xiaolin Hu. Turbo learning framework for human-object interactions recognition and human pose estimation. In *AAAI*, 2019. 2

[6] Chen Gao, Yunpeng Chen, Si Liu, Zhenxiong Tan, and Shuicheng Yan. Adversarialnas: Adversarial neural architecture search for gans. In *CVPR*, 2020. 1

[7] Chen Gao, Si Liu, Ran He, Shuicheng Yan, and Bo Li. Recapture as you want. *arXiv preprint arXiv:2006.01435*, 2020. 1

[8] Chen Gao, Si Liu, Defa Zhu, Quan Liu, Jie Cao, Haoqian He, Ran He, and Shuicheng Yan. Interactgan: Learning to generate human-object interaction. In *ACM MM*, 2020. 1

[9] Chen Gao, Jiarui Xu, Yuliang Zou, and Jia-Bin Huang. Drg: Dual relation graph for human-object interaction detection. In *ECCV*, 2020. 1, 2, 6

[10] Chen Gao, Yuliang Zou, and Jia-Bin Huang. ican: Instance-centric attention network for human-object interaction detection. In *BMVC*, 2018. 1, 2, 6

[11] Ross Girshick. Fast r-cnn. In *CVPR*, 2015. 2

[12] Georgia Gkioxari, Ross Girshick, Piotr Dollár, and Kaiming He. Detecting and recognizing human-object interactions. In *CVPR*, 2018. 2, 6

[13] Saurabh Gupta and Jitendra Malik. Visual semantic role labeling. *arXiv preprint arXiv:1505.04474*, 2015. 2, 5

[14] Tanmay Gupta, Alexander Schwing, and Derek Hoiem. No-frills human-object interaction detection: Factorization, layout encodings, and training techniques. In *ICCV*, 2019. 1, 2, 6

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 6

[16] Zhi Hou, Xiaojiang Peng, Yu Qiao, and Dacheng Tao. Visual compositional learning for human-object interaction detection. In *ECCV*, 2020. 2

[17] Shaofei Huang, Tianrui Hui, Si Liu, Guanbin Li, Yunchao Wei, Jizhong Han, Luoqi Liu, and Bo Li. Referring image segmentation via cross-modal progressive comprehension. In *CVPR*, 2020. 1

[18] Tianrui Hui, Si Liu, Shaofei Huang, Guanbin Li, Sansi Yu, Faxi Zhang, and Jizhong Han. Linguistic structure guided context modeling for referring image segmentation. In *ECCV*, 2020. 1

[19] Wentao Jiang, Si Liu, Chen Gao, Jie Cao, Ran He, Jiashi Feng, and Shuicheng Yan. Psgan: Pose and expression robust spatial-aware gan for customizable makeup transfer. In *CVPR*, 2020. 1

[20] Bumsoo Kim, Taeho Choi, Jaewoo Kang, and Hyunwoo J. Kim. Uniondet: Union-level detector towards real-time human-object interaction detection. In *ECCV*, 2020. 1, 2, 3, 6

[21] Dong-Jin Kim, Xiao Sun, Jinsoo Choi, Stephen Lin, and In So Kweon. Detecting human-object interactions with action co-occurrence priors. In *ECCV*, 2020. 2

[22] H. W. Kuhn. The hungarian method for the assignment problem. In *Naval Research Logistics Quarterly*, 1955. 4

[23] Yong-Lu Li, Xinpeng Liu, Han Lu, Shiyi Wang, Junqi Liu, Jiefeng Li, and Cewu Lu. Detailed 2d-3d joint representation for human-object interaction. In *CVPR*, 2020. 2, 6

[24] Yong-Lu Li, Siyuan Zhou, Xijie Huang, Liang Xu, Ze Ma, Hao-Shu Fang, Yanfeng Wang, and Cewu Lu. Transferable interactiveness knowledge for human-object interaction detection. In *CVPR*, 2019. 1, 2

[25] Yong-Lu Li, Siyuan Zhou, Xijie Huang, Liang Xu, Ze Ma, Hao-Shu Fang, Yan-Feng Wang, and Cewu Lu. Transferable interactiveness prior for human-object interaction detection. In *CVPR*, 2019. 1, 6

[26] Yue Liao, Si Liu, Guanbin Li, Fei Wang, Yanjie Chen, Chen Qian, and Bo Li. A real-time cross-modality correlation filtering method for referring expression comprehension. In *CVPR*, 2020. 1

[27] 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 *CVPR*, 2020. 1, 2, 3, 5, 6, 7

[28] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017. 4

[29] 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 *ECCV*, 2014. 5

[30] Yang Liu, Qingchao Chen, and Andrew Zisserman. Amplifying key cues for human-object-interaction detection. In *ECCV*, 2020. 2, 6

[31] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2018. 6

[32] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, 2016. 2

[33] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In *PMLR*, 2018. 3

[34] Siyuan Qi, Wenguan Wang, Baoxiong Jia, Jianbing Shen, and Song-Chun Zhu. Learning human-object interactions by graph parsing neural networks. In *ECCV*, 2018. 2, 6

[35] Guanghui Ren, Lejian Ren, Yue Liao, Si Liu, Bo Li, Jizhong Han, and Shuicheng Yan. Scene graph generation with hierarchical context. *TNNLS*, 2020. 1

[36] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 2

[37] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *CVPR*, 2019. 4

[38] Liyue Shen, Serena Yeung, Judy Hoffman, Greg Mori, and Li Fei-Fei. Scaling human-object interaction recognition through zero-shot learning. In *WACV*, 2018. 1, 2

[39] Oytun Ulutan, A S M Iftekhar, and B. S. Manjunath. Vsgnet: Spatial attention network for detecting human object interactions using graph convolutions. In *CVPR*, 2020. 6

[40] Bo Wan, Desen Zhou, Yongfei Liu, Rongjie Li, and Xuming He. Pose-aware multi-level feature network for human object interaction detection. In *ICCV*, 2019. 2, 6

[41] Tiancai Wang, Rao Muhammad Anwer, Muhammad Haris Khan, Fahad Shahbaz Khan, Yanwei Pang, Ling Shao, and Jorma Laaksonen. Deep contextual attention for human-object interaction detection. In *ICCV*, 2019. 2

[42] Tiancai Wang, Tong Yang, Martin Danelljan, Fahad Shahbaz Khan, Xiangyu Zhang, and Jian Sun. Learning human-object interaction detection using interaction points. In *CVPR*, 2020. 6

[43] Bingjie Xu, Yongkang Wong, Junnan Li, Qi Zhao, and Mohan S. Kankanhalli. Learning to detect human-object interactions with knowledge. In *CVPR*, 2019. 2

[44] Shan You, Tao Huang, Mingmin Yang, Fei Wang, Chen Qian, and Changshui Zhang. Greedynas: Towards fast one-shot nas with greedy supernet. In *CVPR*, 2020. 1

[45] Fisher Yu, Dequan Wang, Evan Shelhamer, and Trevor Darrell. Deep layer aggregation. In *CVPR*, 2018. 2

[46] Tianyu Yu, Tianrui Hui, Zhihao Yu, Yue Liao, Sansi Yu, Faxi Zhang, and Si Liu. Cross-modal omni interaction modeling for phrase grounding. In *ACM MM*, 2020. 1

[47] Xubin Zhong, Changxing Ding, Xian Qu, and Dacheng Tao. Polysemy deciphering network for human-object interaction detection. In *ECCV*, 2020. 2, 6

[48] Penghao Zhou and Mingmin Chi. Relation parsing neural network for human-object interaction detection. In *ICCV*, 2019. 2

[49] Tianfei Zhou, Wenguan Wang, Siyuan Qi, Haibin Ling, and Jianbing Shen. Cascaded human-object interaction recognition. In *CVPR*, 2020. 6