Human-Object Interaction Detection via Disentangled Transformer

Desen Zhou[1*] Zhichao Liu[1,2*†] Jian Wang[1] Leshan Wang[1,2†] Tao Hu[1] Errui Ding[1] Jingdong Wang[1]
[1]Department of Computer Vision Technology (VIS), Baidu Inc.
[2]ShanghaiTech University
{zhoudesen,wangjian33,hutao06,dingerrui}@baidu.com
{liuzhch,wanglsh}@shanghaitech.edu.cn, wangjingdong@outlook.com

Abstract

Human-Object Interaction Detection tackles the problem of joint localization and classification of human object interactions. Existing HOI transformers either adopt a single decoder for triplet prediction, or utilize two parallel decoders to detect individual objects and interactions separately, and compose triplets by a matching process. In contrast, we decouple the triplet prediction into human-object pair detection and interaction classification. Our main motivation is that detecting the human-object instances and classifying interactions accurately needs to learn representations that focus on different regions. To this end, we present Disentangled Transformer, where both encoder and decoder are disentangled to facilitate learning of two subtasks. To associate the predictions of disentangled decoders, we first generate a unified representation for HOI triplets with a base decoder, and then utilize it as input feature of each disentangled decoder. Extensive experiments show that our method outperforms prior work on two public HOI benchmarks by a sizeable margin. Code will be available.

1. Introduction

Human-object interaction(HOI) detection [11] aims at detecting all the <human, verb, object> triplets in an image. It has attracted increasing attention in the computer vision community in recent years [8, 10]. Accurate estimation of human-object interactions can benefit multiple downstream tasks, such as human action recognition [38], scene graph generation [25], and image caption [4].

Recent advances show that HOI detection can be formulated as set prediction problem [3, 17, 30, 44]. Existing HOI transformers can be categorized into two types: single-branch transformer and parallel-branch transformer. Single-branch transformer [30, 44] adopts multi-task strategy, in which one query is responsable for predicting a <human, verb, object> triplet within a single decoder. In contrast, parallel-branch transformer [3, 17] adopts parallel decoders for instance detection and interaction classification separately. Specifically, one instance decoder follows DETR [1] and detects individual objects, and the other interaction decoder estimates the interactions in the image. To compose HOI triplets, it generates additional associative embeddings to match the interactions and instances. Since HOI detection is a composition problem [13, 15], the latter decomposing strategy has several advantages compared with unified multi-tasking strategy. First, two sub-task decoders

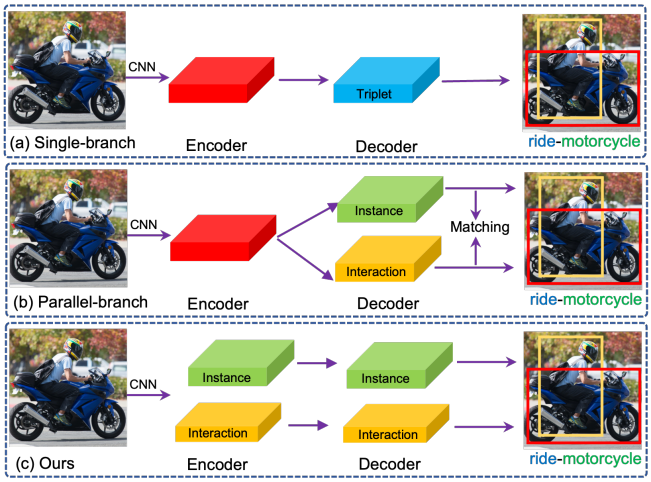


Figure 1. Architecture comparison of different HOI transformers. (a) Single-branch transformer [30, 44] adopts a single decoder to directly detect HOI triplets. (b) Parallel-branch transformer [3, 17] utilizes separate decoders detect individual objects and interactions, and then compose triples by a matching process, which might introduce additional grouping errors. (c) Ours disentangles the task of triplet prediction into human-object pair detection and interaction classification via an instance stream and an interaction stream, where both encoder and decoder are disentangled.

*Equal contribution.
†Work done when Zhichao and Leshan were interns at VIS, Baidu.

might attend to different regions via cross attention to facilitate learning and also results in better interpretability. In addition, it has better generalizability, especially for rare categories due to long-tail distribution of triplet compositions. However, existing parallel-decoder transformers suffer from two crucial drawbacks under complex scenarios: i) the interaction predictions have to find their corresponding human and object instances in instance decoder, which might introduce additional errors due to mis-grouping; ii) regardless of the shared encoder, the decoding sub-tasks are relatively independent and the joint configurations of instances and interactions are not considered.

To overcome above limitations, we present Disentangled Transformer(DisTR). We decouple the triplet prediction into human-object pair detection and interaction classification via an instance stream and an interaction stream, where both encoder and decoder are disentangled. An illustration of architecture comparison between ours and prior HOI transformers is shown in Fig.1. Our encoder module extracts different contextual information for two subtasks. During decoding process, the task decoder decodes its representation based on the corresponding task encoder. Different from prior parallel-decoder transformers [3, 17] that the instance decoder predicts individual objects, our instance decoder predicts a set of interactive human-object pairs. To associate the predictions of task decoders, we adopt a base decoder to first generate a unified representation for HOI triplets, following QPIC [30], and then utilize it as input feature of each task decoder. The task decoder then refines its representation based on the unified representation, resulting in a coarse-to-fine process. We further design an attentional fusion block to pass information between task decoders help them communicate with each other.

We evaluate our proposed method on two public benchmarks: V-COCO [11] and HICO-DET [2]. Our method outperforms current state-of-the-art by a sizeable margin. We further visualize the cross attentions in our task decoders, and observe that our task decoders indeed attend to different spatial regions, demonstrating the effectiveness of our proposed disentangled strategy.

The contributions of this paper are three folds:

- We propose a disentangled strategy for HOI detection, where the triplet prediction is decoupled into human-object pair detection and interaction classification via an instance stream and an interaction stream.

- We develop a new transformer, where both encoder and decoder are disentangled. We also propose a coarse-to-fine strategy to associate the predictions of instance decoder and interaction decoder, and an attentional fusion block for communication between task decoders.

- We achieve new state-of-the-art on both V-COCO and HICO-DET benchmarks.

2. Related Work

2.1. Two-stage Methods

A classical branch of research to HOI detection are based on the hypothesis-and-classify strategy, which first detects object instances via object detectors [9,29], and then perform interaction classification on the grouped pairwise human-object proposals [8, 10, 21, 22, 26, 32]. Some works also exploit graph structure to enhance object dependencies [27, 28, 31, 33, 40]. Another bunch of two-stage methods is the compositional approaches [13–15, 20], which disentangle HOI representations by learning from fabricated compositional HOIs. In contrast, our method disentangles representations by disentangled task encoders and decoders and its one-stage framework does not rely on pre-computed object proposals.

2.2. One-stage Methods

Recently, one-stage or parallel HOI has caused extensive concern which transforms the interaction target as a center point or interaction object, and then adopt a detection pipeline. PPDM [23] which is based on CenterNet [5] detects the interaction centers as well as objects, and then perform grouping as its post-process. IP-Net [35] is similar. UnionDet [16] use a novel union-level detector that eliminates this additional inference stage by directly capturing the region of interaction. DIRV [6] concentrates on the densely sampled interaction regions across different scales for each human-object pair and introduce a novel voting strategy to replace Non-Maximal Suppression(NMS).

HOI Transformer Recent HOI transformers follow DETR [1], but separate into two types: entangled transformer and disentangled transformer. The entangled transformer, QPIC [30] and HOITrans [44] directly generate multiple <human,object,action> triples of given image with a single decoder. On the contrary, disentangled transformers, HOTR [17] and ASNet [3] predict the objects and interactions in parallel decoders, and then perform matching between objects and interaction targets to generate final predictions. Recently, Zhang et.al [39] propose to disentangle the instance decoder and interaction decoder in a cascaded process, which treats the instance decoder as proposal generator to interaction decoder. In contrast, our sub-tasks are parallelly decoded, hence the communication can be applied. In addition, our disentanglement is more complete due to encoder disentanglement.

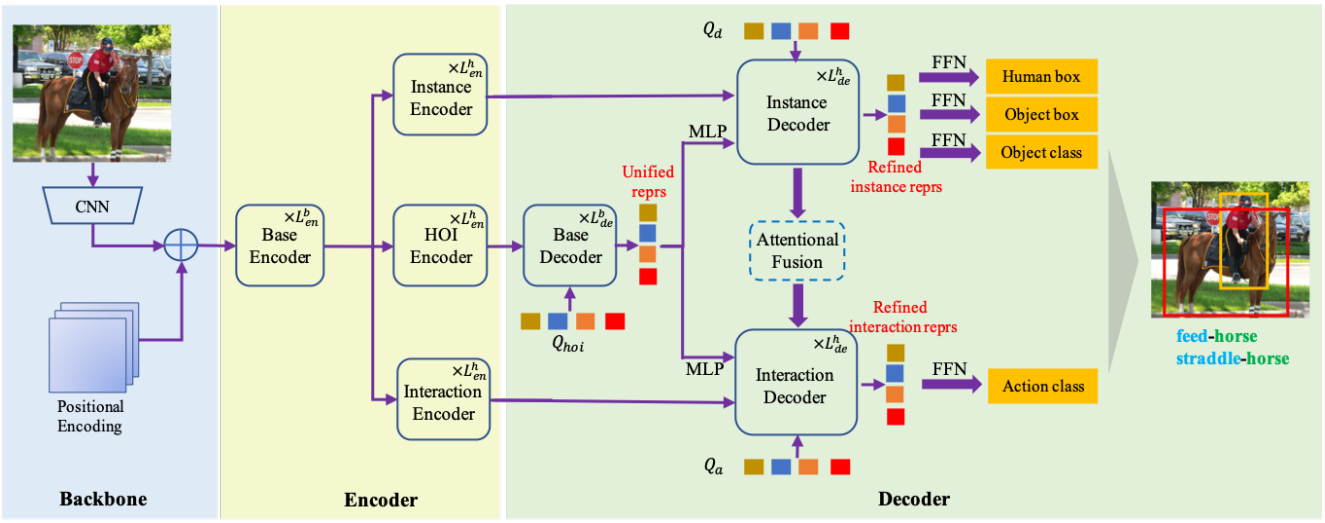


Figure 2. Overview of our framework. **Encoder module** extracts global contexts at three levels for different decoding sub-tasks. **Decoder module** disentangles the representations of instances and interactions in a coarse-to-fine manner: the *base decoder* extracts unified HOI representation of HOI triplets, then the *instance decoder* and *interaction decoder* refines the unified HOI representation in disentangled feature spaces. Our instance decoder directly estimates interactive human-object instance pairs, which are associated with interaction predictions. The *Attentional fusion blocks* are further inserted at each output layer(excluding the last layer) of two task decoders to perform communication between them.

3. Method

3.1. Overview

We adopt the one-stage transformer framework, which directly estimates all the <human,verb,object> triplets given an image. To achieve this, we first group the HOI triplets with the same human and object instances. Then, the ground truth of an image can be represented as a tuple set $\{(\tilde{\mathbf{x}}_h^i, \tilde{\mathbf{x}}_o^i, \tilde{\mathbf{c}}^i, \tilde{\mathbf{a}}^i)|i = 1, 2, ..., M\}$, where M is the number of ground truth human-object interaction pairs, $\tilde{\mathbf{x}}_h^i, \tilde{\mathbf{x}}_o^i \in \mathbb{R}^4$ denote the bounding boxes of human instance and object instance, $\tilde{\mathbf{c}} \in \{0, 1\}^C$ indicates the one-hot encoding of object category and C is the number of object classes, $\tilde{\mathbf{a}}^i \in \{0, 1\}^{\mathcal{A}}$ denotes the labels of $\mathcal{A}$ interaction classes. We then deploy our transformer network to predict such tuple set. Formally, given image I, our goal is to define a transformer network $\mathcal{F}$ that performs the mapping:

$$I \xrightarrow{\mathcal{F}} \{(\mathbf{x}_h^i, \mathbf{x}_o^i, \mathbf{c}^i, \mathbf{a}^i)|i = 1, 2, ..., N_q\}, \tag{1}$$

where i is the query index and N_q is the number of queries pre-defined in our transformer network. $\mathbf{x}_h^i, \mathbf{x}_o^i \in \mathbb{R}^4$ denote the predicted bounding boxes of human instance and object instance respectively, $\mathbf{c}_o^i \in (0, 1)^{C+1}$ is estimated probability of object classification, which is normalized by softmax function. The additional dimension indicates background non-object class. $\mathbf{a}^i \in (0, 1)^{\mathcal{A}}$ indicates the interaction probabilities, which are normalized by sigmoid function.

We adopt a coarse-to-fine strategy to disentangle the instance detection and interaction classification, to resolve the matching problem between predictions. Specifically, we first generate a unified HOI representation to represent the HOI triplets $\{(\mathbf{x}_h^i, \mathbf{x}_o^i, \mathbf{c}^i, \mathbf{a}^i)\}$, then an instance decoder is utilized to refine the representation in instance space and predict the human-object instance pairs, indicated by $\{(\mathbf{x}_h^i, \mathbf{x}_o^i, \mathbf{c}^i)\}$. And the interaction decoder is responsable for interaction disentanglement and prediction, indicated by $\{\mathbf{a}^i\}$. During inference, the predictions of the same query index in two head decoders are directly grouped together. Below we introduce our detailed implementation of above coarse-to-fine disentangling strategy.

3.2. Network Architecture

Similar to existing HOI transformers [3, 30] and DETR [1], our network consists of three main modules: backbone module computes image features; encoder module exploits self-attention mechanism to further extract higher relational contexts between different spatial regions; and decoder module extracts representations from encoder module for the disentangled sub-tasks of instance detection and interaction classification. An overview of our framework is shown in Fig.2.

3.2.1 Backbone module

A CNN backbone is used to extract the high level semantic feature map with shape (H, W, C), and then a 1×1 convolution layer is used to reduce the channel dimension from C to D. We flatten the feature map of shape (H, W, D) to

(HW, D). We utilize ResNet50 [12] as our backbone, and reduce the feature map in conv-5 using 1×1 convolution from $C = 2048$ to $D = 256$, the backbone visual features are represented as $\Gamma_{back} \in \mathbb{R}^{HW \times D}$.

3.2.2 Encoder module

Our encoder module aims at modeling relationships at different spatial regions to enhance global contexts for backbone representation Γ_{back}. Prior parallel-decoder transformers [3, 17] utilize shared encoder for instance detection and interaction classification. However, we assume that the relations in image representations of different subtasks are different and the encoder representations better be designed for specific sub-tasks. Hence we disentangle our encoder at three levels for different decoding sub-tasks: human-object pair detection, interaction classification and unified representation generation. Specifically, it consists of a base encoder and three head encoders. The base encoder, which consists of L^b_{en} layers, enhances Γ_{back} to generate a base encoder representation Γ^b_{en}. Then, three different head encoders with L^h_{en} layers refine the base encoder representation separately. We denote the refined head representations as $\Gamma^{hoi}_{en}, \Gamma^d_{en}, \Gamma^a_{en}$, which are used for computing cross attentions in different decoders: Γ^{hoi}_{en} for base decoder, Γ^d_{en} for instance decoder and Γ^a_{en} for interaction decoder. All the encoder representations share the same shape: $\Gamma^b_{en}, \Gamma^{hoi}_{en}, \Gamma^d_{en}, \Gamma^a_{en} \in \mathbb{R}^{HW \times D}$.[1]

3.2.3 Decoder module

Our decoder module adopts attention mechanism to extract representations from encoder for sub-task decoding. We disentangle the representations of instances and interactions in a coarse-to-fine manner, which first utilizes a base decoder to generate a unified representation for a HOI triplet, and then exploits another two disentangled decoders to refine the unified representation in the spaces of instances and interactions. Different from previous transformers [3, 17] that the instance decoder predicts individual objects regardless of their interactiveness, our instance decoder estimates interactive human-object instance pairs associated with the interaction prediction. Hence it requires no additional matching process. To further help two task decoders communicate with each other, we propose an attentional fusion block to pass information between them. Below we describe the detailed structures of above components.

Base decoder Our base decoder has L^b_{de} layers and generates unified HOI representations for the disentangled decoders to facilitate feature refinements and associate predictions. Formally, the base decoder $\mathcal{F}^b_{de}$ transforms a set of learnable HOI queries $Q_{hoi} \in \mathbb{R}^{N_q \times D}$ into a set of base HOI representations $\Gamma^b_{de} \in \mathbb{R}^{N_q \times D}$ from HOI encoder head:

$$\Gamma^b_{de} = \mathcal{F}^b_{de}(\mathbb{0}, \Gamma^{hoi}_{en}, \mathbf{p}_{en}, Q_{hoi}), \tag{2}$$

where the zero matrix $\mathbb{0} = \{0\}^{N_q \times D}$ indicates the input feature of base decoder. $\mathbf{p}_{en} \in \mathbb{R}^{HW \times D}$ is the position embedding of the encoder representations.

Instance decoder Our instance decoder aims at refining the unified HOI representation Γ^b_{de} to generate a disentangled representation for interactive human-object instance pairs. To achieve this, we utilize a MLP to embed the unified representation to generate input feature of instance decoder. Our instance decoder $\mathcal{F}^d_{de}$ has L^h_{de} layers, and takes the input feature, together with a set of learnable instance queries $Q_d \in \mathbb{R}^{N_q \times d}$ to perform feature refinement. We found that inputting the unified representation as decoder feature is better than directly utilizing it as queries, because the disentangled decoders will have a powerful initial feature. The output of the instance decoder is a set of interactive human-object instance pairs:

$$\{(\mathbf{x}^i_h, \mathbf{x}^i_o, \mathbf{c}^i)\} = \mathcal{F}^d_{de}(\text{MLP}(\Gamma^b_{de}), \Gamma^d_{en}, \mathbf{p}_{en}, Q_d). \tag{3}$$

Interaction decoder Similar to the instance decoder, our L^h_{de}-layer interaction decoder refines the unified HOI representation to the disentangled interaction feature space and generate a set of interaction predictions:

$$\{\mathbf{a}^i\} = \mathcal{F}^a_{de}(\text{MLP}(\Gamma^b_{de}), \Gamma^a_{en}, \mathbf{p}_{en}, Q_a), \tag{4}$$

where $Q_a \in \mathbb{R}^{N_q \times D}$ indicates the query set, Γ^a_{en} is the representation of interaction encoder. Similar to instance decoder, during decoding, the estimated interactions are associated with unified HOI representation, as well as the human-object pairs in instance decoder.

Attentional fusion block Our disentangled task decoders perform sub-tasks separately. However, two functional modules are not sufficiently communicated due to early decomposition of unified representations.[2] To make the subtasks better benefit from each other, we perform message passing between the instance decoder and interaction decoder. Specifically, in the output of each layer in disentangled decoders, we fuse the instance representation to the interaction representation if they are associated with the same query index. The design of our fusion block is inspired by [37] which utilizes the object representation and action representation to estimate a channel attention. Formally, we

[1] In this section, 'b' is short for base, 'd' indicates detection/instance, 'a' indicates action/interaction, 'h' indicates head.

[2] In our model, the instance decoder and interaction decoder have more layers than the base decoder.

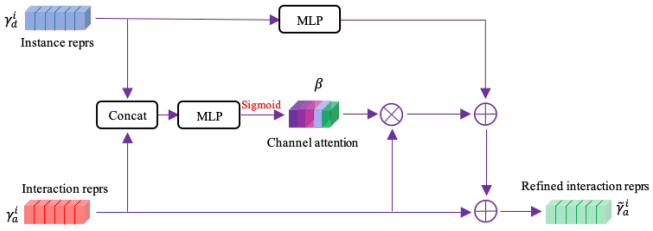


Figure 3. The structure of attentional fusion block.

denote the instance representation and interaction representation for query i as $\gamma_d^i, \gamma_a^i \in \mathbb{R}^D$. As shown in Fig.3, our attentional fusion block first concatenates the γ_d^i and γ_a^i and compute a channel attention $\beta \in \mathbb{R}^D$ with a MLP:

$$\beta = \sigma(\text{MLP}(\text{Concat}([\gamma_a^i; \gamma_d^i]))), \tag{5}$$

where σ is the sigmoid function to constrain the elements in β to range $(0, 1)$. The channel attention is used to enhance the interaction representation with element-wise multiplication. During practice, we found that adding instance features provides additionally improvement. Hence, the output interaction representation $\tilde{\gamma}_a^i \in \mathbb{R}^D$ has the form:

$$\tilde{\gamma}_a^i = \gamma_a^i + \beta \odot \gamma_a^i + \text{MLP}(\gamma_d^i), \tag{6}$$

where $\odot$ indicates the element-wise multiplication. In the last layer of disentangled decoders, we do not apply attentional fusion, in order to make the final representations more discriminative.

3.3. Model Learning

We adopt similar losses with previous HOI Transformer [30, 44]. Specifically, the instance decoder and interaction decoder generates set predictions $\{(\mathbf{x}_h^i, \mathbf{x}_o^i, \mathbf{c}^i)\}$ and $\{\mathbf{a}^i\}$, where $\mathbf{c}^i \in (0,1)^{C+1}$, $\mathbf{a}^i \in (0,1)^{\mathcal{A}}$ indicate the object class probabilities and interaction class probabilities, which are normalized by softmax and sigmoid respectively. Then the predictions with the same query index are grouped together to a triplet set $\{(\mathbf{x}_h^i, \mathbf{x}_o^i, \mathbf{c}^i, \mathbf{a}^i)\}$. The rest process is the same as previous HOI transformers [30] that first exploit the combined triplet predictions to compute a Hungarian Matching to the ground truth triplets, and then adopt different loss functions to the matched triplets. We denote $\mathcal{L}_b, \mathcal{L}_u, \mathcal{L}_c, \mathcal{L}_a$ as bounding box $L1$ losses, GIoU loss, object classification loss and interaction classification loss, the overall loss given by:

$$\mathcal{L} = \lambda_b \mathcal{L}_b + \lambda_u \mathcal{L}_u + \lambda_c \mathcal{L}_c + \lambda_a \mathcal{L}_a, \tag{7}$$

where $\lambda_b, \lambda_u, \lambda_c, \lambda_a$ denote the weights to balance the different loss components.

Auxiliary loss Inspired by DETR [1], we add prediction FFNs and adopt auxiliary losses to each decoder layer to extract better representations. Our base decoder decodes unified HOI representations and predicts HOI triplets. The disentangled decoders predict instances and interactions respectively. Since the representations in disentangled decoders are refined from unified representation, we adopt different prediction FFNs to the disentangled decoders and base decoder. While in the same decoder, FFN parameters are shared.

3.4. Model Inference

Given HOI prediction set $\{(\mathbf{x}_h^i, \mathbf{x}_o^i, \mathbf{c}^i, \mathbf{a}^i)\}$, where $\mathbf{c}^i \in (0,1)^{C+1}$, $\mathbf{a}^i \in (0,1)^{\mathcal{A}}$ denote the classification probabilities of object class and action classes, the predicted object class and its detection score is given by $\text{argmax}_k \mathbf{c}_k^i$ and $\max_k \mathbf{c}_k^i$, the output HOI of j-th action in i-th query is given by $(\mathbf{x}_h^i, \mathbf{x}_o^i, \text{argmax}_k \mathbf{c}_k^i, j)$ with a prediction score $\max_k \mathbf{c}_k^i \cdot \mathbf{a}_j^i$. Similar to prior work [30], we only keep a prediction if its confidence score is above a threshold.

4. Experiments

4.1. Experimental Setup

Dataset We conducted experiments on two HOI detection datasets: HICO-DET [2] and V-COCO [11]. V-COCO is derived from MS-COCO [24] and contains 5400 and 4946 images in trainval subset and test subset respectively. V-COCO is annotated with 80 object categories and 29 action classes including 25 HOI triplets and 4 human body actions. HICO-DET contains 38118 and 9658 images for training and testing respectively. The HICO-DET has 80 object categories which is same as MS-COCO and 117 verb categories, all objects and verbs consist of 600 HOI triplets.

Evaluation Metrics Following prior work [3, 8, 30], we use mean average precision(mAP). A triplets prediction is considered positive if human and object boxes have a IOU larger than 0.5 with ground truth boxes, and the predicted object categories and verb categories need to be correct. For HICO-DET, we report mAP over Full, Rare, and Non-Rare settings. For V-COCO, we report mAP on scenario #1 (including objects) and scenario #2 (ignore objects).

4.2. Implementation Details

In our implementation, the layer number of base encoder, head encoder, base decoder and head decoder are set to $L_{en}^b = 4$, $L_{en}^h = 2$, $L_{de}^b = 2$, $L_{de}^h = 4$. Query number $N_q = 100$. We set the weight coefficients of $\lambda_b, \lambda_u, \lambda_c, \lambda_a$ to 2.5, 1, 1, 1. During training, we initialize our model parameters with pre-trained DETR [1] on COCO dataset. For the missing parameters, we adopt a warmup strategy, which first freezes the pre-trained parameters and adjusts the missing parameters for 10 epochs. Following prior work [3, 30], we set the parameters in encoder and decoder to 10^{-4}, and

Method	Backbone	Scenario #1	Scenario #2
Two-stage Method			
iCAN [8]	R50	45.3	52.4
TIN [22]	R50	47.8	54.2
VCL [13]	R101	48.3	-
DRG [7]	R50-FPN	51.0	-
VSGNet [31]	R152	51.8	57.0
PMFNet [32]	R50-FPN	52.0	-
PDNet [41]	R152	52.6	-
CHGNet [33]	R50	52.7	-
FCMNet [26]	R50	53.1	-
ACP [18]	R152	53.2	-
IDN [20]	R50	53.3	60.3
SCG [40]	R50-FPN	54.2	60.9
One-stage Method			
UnionDet [16]	R50-FPN	47.5	56.2
IPNet [35]	HG104	51.0	-
GG-Net [42]	HG104	54.7	-
DIRV [6]	EfficientDet-d3	56.1	-
HOITrans [44]	R101	52.9	-
AS-Net [3]	R50	53.9	-
HOTR [17]	R50	55.2	64.4
QPIC [30]	R50	58.8	61.0
Ours	R50	**66.2**	**68.5**

Table 1. Performance comparison on V-COCO test set.

Method	Scenario #1	Default(Full)
Ours	**66.2**	**31.75**
w/o encoder disentanglement	65.5	30.79
w/o attentional fusion	64.4	31.24
w/o decoder disentanglement	58.8	29.07

Table 2. Ablation study of model components on both V-COCO test set (Scenario #1) and HICO-DET test set (Default, Full setting)

the backbone to 10^{-5}. Weight decay is set to 10^{-4}. Batch size is set to 16. For V-COCO, we freeze the backbone to avoid over-fitting. For HICO-DET, we fine-tune the whole model end-to-end. Including warmup, HICO-DET and V-COCO are trained with 80 epochs and learning rate is decreased at 65th epoch with 10 times. Our experiments are conducted on 8 Tesla V100 GPUs.

4.3. Comparison to State-of-the-art

We show the comparison of our method with previous two-stage and one-stage methods in Tab. 1 and Tab. 3. Our method outperforms prior works on both benchmarks.

On V-COCO dataset, compared with state-of-the-art one-stage method QPIC [30], ours outperforms it with a significant gap. Compared with state-of-the-art two-stage method SCG [40], our method also yields a large performance gap with 12.0% mAP. It illustrates the our method has overwhelming advantage on both one-stage and two-stage methods. Particularly, our method outperforms previous parallel-branch HOI transformer HOTR [17] and AS-Net [3] by a large margin with 11.0% mAP and 12.3% mAP under scenario #1.

On HICO-DET dataset, compared with state-of-the-art one-stage methods, with R50 backbone, our method outperforms QPIC [30] by 2.68% mAP, and AS-Net [3] by 2.88% mAP under Default Full setting. It's also worth noting that under Rare setting, our method achieves 27.45%, which is significant better than QPIC, demonstrating the effectiveness of disentangled strategy. Our method also outperforms recent state-of-the-art two-stage method SCG [38] by 0.42% map. However, the two stage pipeline includes heuristic processes such as NMS and is not end-to-end.

4.4. Ablation Study

w/o encoder disentanglement Our model adopts a disentangled encoder to extract global contexts at three levels for different decoding sub-tasks. We replace the disentangled encoder with a single encoder of same layer in our full model, the performance drops 0.7% mAP and 0.96% mAP on both V-COCO and HICO-DET datasets respectively, as shown in Tab. 2.

w/o attentional fusion Our attentional fusion block provides communications between two task decoders. As shown in Tab. 2, we remove the attentional fusion block, the performance drops 1.8% mAP and 0.51% mAP on V-COCO and HICO-DET datasets respectively.

w/o decoder disentanglement Our disentangled decoder is the key in our framework. It predicts interactive human-object instance pairs instead of individual objects as in prior parallel-branch transformers [3, 17], and exploits unified HOI representation to associate instances and interactions. Without our decoder disentanglement, our model is more like the QPIC [30]. Hence we compare the performances of ours and the single-branch transformer in Tab. 2. We can observe that performances significantly drop on both datasets.

Effect of warmup strategy Since our transformer model has more parameters than original DETR, we adopt a warmup strategy during training. To validate the effectiveness of our warmup strategy, we perform an ablation study about the warmup strategy, shown in Tab.6. We notice that the warmup strategy slightly improves the performances on both datasets.

Different layers of base/head encoders/decoders We further perform ablation study on different transformer layers of base encoder/decoder and disentangled head encoders/decoders, shown in Tab.4. For simplicity of our model and usage of pre-trained DETR parameters, we empirically keep the sum of base layer and head layer to 6, as in the original transformer. From the first three rows, we can observe that the decoder base layer $L^b_{de} = 2$ and head layer $L^h_{de} = 4$ is the best proportion and provides best

Method	Detector	Backbone	Default			Known Object		
			Full	Rare	Non-Rare	Full	Rare	Non-rare
Two-stage Method								
GPNN [28]	COCO	R101	13.11	9.34	14.23	-	-	-
iCAN [8]	COCO	R50	14.84	10.45	16.15	16.26	11.33	17.73
DCA [34]	COCO	R50	16.24	11.16	17.75	17.73	12.78	19.21
TIN [22]	COCO	R50	17.03	13.42	18.11	19.17	15.51	20.26
RPNN [43]	COCO	R50	17.35	12.78	18.71	-	-	-
PMFNet [32]	COCO	R50-FPN	17.46	15.65	18.00	20.34	17.47	21.20
FCMNet [26]	COCO	R50	20.41	17.34	21.56	22.04	18.97	23.12
DJ-RN [19]	COCO	R50	21.34	18.53	22.18	23.69	20.64	24.60
IDN [20]	COCO	R50	23.36	22.47	23.63	26.43	25.01	26.85
VCL [13]	HICO-DET	R50	23.63	17.21	25.55	25.98	19.12	28.03
DRG [7]	HICO-DET	R50-FPN	24.53	19.47	26.04	27.98	23.11	29.43
IDN [20]	HICO-DET	R50	24.58	20.33	25.86	27.89	23.64	29.16
SCG [40]	HICO-DET	R50-FPN	31.33	24.72	33.31	34.37	27.18	36.52
One-stage Method								
UnionDet [16]	HICO-DET	R50-FPN	17.58	11.72	19.33	19.76	14.68	21.27
IPNet [35]	COCO	R50-FPN	19.56	12.79	21.58	22.05	15.77	23.92
PPDM [23]	HICO-DET	HG104	21.94	13.97	24.32	24.81	17.09	27.12
DIRV [6]	HICO-DET	EfficientDet-d3	21.78	16.38	23.39	25.52	20.84	26.92
HOTR [17]	HICO-DET	R50	25.10	17.34	27.42	-	-	-
HOITrans [44]	HICO-DET	R101	26.61	19.15	28.84	29.13	20.98	31.57
AS-Net [3]	HICO-DET	R50	28.87	24.25	30.25	31.74	27.07	33.14
QPIC [30]	HICO-DET	R50	29.07	21.85	31.23	31.68	24.14	33.93
QPIC [30]	HICO-DET	R101	29.90	23.92	31.69	32.38	26.06	34.27
Ours	HICO-DET	R50	**31.75**	**27.45**	**33.03**	**34.50**	**30.13**	**35.81**

Table 3. Performance comparison on HICO-DET. 'COCO' means the object detector is freeze and pretrained on MS-COCO, 'HICO-DET' means the model is fine-tuned on HICO-DET training set.

	base	head	Scenario #1	Scenario #2
Decoder	1	5	65.6	67.5
	2	4	**66.2**	**68.5**
	3	3	64.7	66.5
Encoder	5	1	65.6	67.6
	4	2	**66.2**	**68.5**
	3	3	65.1	67.1

Table 4. Ablation study on different transformer layers of base encoder/decoder and disentangled head encoders/decoders on VCOCO test set.

Method	VCOCO	HICO
feature decomposition(proposed)	**66.2**	**31.75**
query decomposition	64.9	31.09

Table 5. Different association strategies of instances and interactions on V-COCO test set (Scenario #1) and HICO-DET test set (Default, Full setting)

Method	VCOCO	HICO
w/o warmup	65.7	31.49
w/ warmup	**66.2**	**31.75**

Table 6. Effect of warmup strategy on V-COCO test set (Scenario #1) and HICO-DET test set (Default, Full setting)

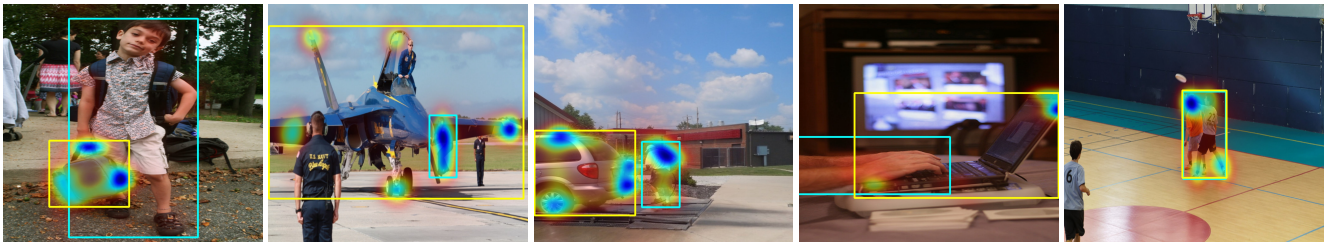
performance, demonstrating the importance of unified representation. From the bottom three rows, we can see that 4-layer base with 2-layer head outperforms 3-layer base with 3-layer head, which implies that the modeling of shared global contexts in base encoder is also important.

Different association strategies Different from previous parallel-branch HOI transformer [3, 30] that instance decoder predicts individual objects in the image, our instance decoder directly estimates a set of interactive human-object instance pairs. In our framework, we adopt a base decoder to generate a unified representation to associate the estimated human-object instance pairs and interactions. We notice that there might be different association strategies. To study the effectiveness of our coarse-to-fine association strategy(referred to as feature decomposition), we replace the unified representation with a set of learnable unified HOI queries, which are then used to generate two queries with MLPs for disentangled decoders(referred to as query decomposition). We keep our disentangled encoder and attentional fusion block for fair comparison. As shown in Tab. 5, the performance drops by 1.3% mAP and 0.66% mAP on V-COCO and HICO-DET datasets respectively, which implies our association strategy is effective.

4.5. Model Complexity Analysis

Since our model includes more encoder/decoder and fusion blocks, readers may care about the complexity of our

(a) cross attention in interaction decoder



hold-handbag board-airplane wash-car work on computer instr-laptop throw-frisbee

(b) cross attention in instance decoder

Figure 4. Visualization of cross attention maps of the same triplet prediction in our interaction decoder(top row) and instance decoder(bottom row). The left three samples are from HICO-DET [2] and others are from V-COCO [11]. In the top row, we can see that our interaction decoder attends to the interactive regions of human and objects. In the bottom row, we can see that our instance decoder attends to the object extremities. The different regions the model attends to implies that interaction and instance decoders indeed capture the disentangled representations of images.

Method	Backbone	AP	Params(M)	FLOPs(G)
QPIC [30]	R50	58.8	41.68	87.87
QPIC [30]	R101	58.3	60.62	156.18
AS-Net [3]	R50	53.9	52.75	88.86
HOTR [17]	R50	55.2	51.41	88.78
HOITrans [44]	R101	52.9	60.62	156
Ours	R50	**66.2**	**57.31**	**94.23**

Table 7. Model complexity comparison between ours and prior state-of-the-art HOI transformers. 'AP' indicates the performances on V-COCO test set under scenario #1.

model. Therefore, we compare the parameters and FLOPS of our final model and prior HOI Transformers in Tab .7. Similar to DETR [1], we compute the FLOPS with the tool **flop_count_operators** from Detectron2 [36] for the first 100 images in the V-COCO test set and calculate the average numbers. We observe that our model has comparable parameters and FLOPS compared with prior HOI transformers. In particular, our model merely introduces 7% extra FLOPS compared with the single-branch QPIC under R50, demonstrating both efficiency and effectiveness of our disentangled transformer.

4.6. Qualitative Analysis

As shown in Fig 4, we visualize the cross attention maps of the same triplet prediction in instance decoder and interaction decoder. Top row shows the attention maps of interaction decoder, we can observe that the attention maps highlight the interactive regions between human-object instance pairs. In the bottom row, we can observe that the instance attention map attends to the object extremities, which is similar to DETR [1]. The different attention maps implies that our instance and interaction decoders indeed capture disentangled representations.

5. Conclusion

In this paper, we propose disentangled transformer for HOI detection. Our method decouples the triplet prediction into human-object pair detection and interaction classification via an instance stream and an interaction stream, where both encoder and decoder are disentangled. To associate the predictions of two task decoders, we adopt a coarse-to-fine strategy that first utilizes a base decoder to generate a unified HOI representation, and then conduct feature refinement in the disentangled instance and interaction spaces. We further propose an attentional fusion block to help two task decoders communicate with each other. As a result, our method is able to outperform prior HOI transformers and other methods by a sizeable margin on both V-COCO and HICO-DET benchmarks. The visualization of cross attention maps in task decoders also provide a good interpretation of the disentangled strategy.

Potential Negative Societal Impact

Our algorithm has no evident threats to society. However, someone might use our method for malicious usage, e.g. to attack people in military usage or invasion of privacy with surveillance. Therefore, we encourage good faith consideration before adopting our technology.

References

[1] 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. 1, 2, 3, 5, 8

[2] Yu-Wei Chao, Yunfan Liu, Xieyang Liu, Huayi Zeng, and Jia Deng. Learning to detect human-object interactions. In *2018 ieee winter conference on applications of computer vision (wacv)*, pages 381–389. IEEE, 2018. 2, 5, 8

[3] Mingfei Chen, Yue Liao, Si Liu, Zhiyuan Chen, Fei Wang, and Chen Qian. Reformulating hoi detection as adaptive set prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9004–9013, 2021. 1, 2, 3, 4, 5, 6, 7, 8

[4] Shizhe Chen, Qin Jin, Peng Wang, and Qi Wu. Say as you wish: Fine-grained control of image caption generation with abstract scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9962–9971, 2020. 1

[5] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Keypoint triplets for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6569–6578, 2019. 2

[6] Hao-Shu Fang, Yichen Xie, Dian Shao, and Cewu Lu. Dirv: Dense interaction region voting for end-to-end human-object interaction detection. *arXiv preprint arXiv:2010.01005*, 2020. 2, 6, 7

[7] Chen Gao, Jiarui Xu, Yuliang Zou, and Jia-Bin Huang. Drg: Dual relation graph for human-object interaction detection. In *European Conference on Computer Vision*, pages 696–712. Springer, 2020. 6, 7

[8] Chen Gao, Yuliang Zou, and Jia-Bin Huang. ican: Instance-centric attention network for human-object interaction detection. *arXiv preprint arXiv:1808.10437*, 2018. 1, 2, 5, 6, 7

[9] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 2

[10] 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*, pages 8359–8367, 2018. 1, 2

[11] Saurabh Gupta and Jitendra Malik. Visual semantic role labeling. *arXiv preprint arXiv:1505.04474*, 2015. 1, 2, 5, 8

[12] 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. 4

[13] Zhi Hou, Xiaojiang Peng, Yu Qiao, and Dacheng Tao. Visual compositional learning for human-object interaction detection. In *European Conference on Computer Vision*, pages 584–600. Springer, 2020. 1, 2, 6, 7

[14] Zhi Hou, Baosheng Yu, Yu Qiao, Xiaojiang Peng, and Dacheng Tao. Affordance transfer learning for human-object interaction detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 495–504, 2021. 2

[15] Zhi Hou, Baosheng Yu, Yu Qiao, Xiaojiang Peng, and Dacheng Tao. Detecting human-object interaction via fabricated compositional learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14646–14655, 2021. 1, 2

[16] Bumsoo Kim, Taeho Choi, Jaewoo Kang, and Hyunwoo J Kim. Uniondet: Union-level detector towards real-time human-object interaction detection. In *European Conference on Computer Vision*, pages 498–514. Springer, 2020. 2, 6, 7

[17] 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. 1, 2, 4, 6, 7, 8

[18] Dong-Jin Kim, Xiao Sun, Jinsoo Choi, Stephen Lin, and In So Kweon. Detecting human-object interactions with action co-occurrence priors. In *European Conference on Computer Vision*, pages 718–736. Springer, 2020. 6

[19] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10166–10175, 2020. 7

[20] Yong-Lu Li, Xinpeng Liu, Xiaoqian Wu, Yizhuo Li, and Cewu Lu. Hoi analysis: Integrating and decomposing human-object interaction. *Advances in Neural Information Processing Systems*, 33:5011–5022, 2020. 2, 6, 7

[21] Yong-Lu Li, Liang Xu, Xinpeng Liu, Xijie Huang, Yue Xu, Shiyi Wang, Hao-Shu Fang, Ze Ma, Mingyang Chen, and Cewu Lu. Pastanet: Toward human activity knowledge engine. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 382–391, 2020. 2

[22] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3585–3594, 2019. 2, 6, 7

[23] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 482–490, 2020. 2, 7

[24] 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 *European conference on computer vision*, pages 740–755. Springer, 2014. 5

[25] 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*, pages 3746–3753, 2020. 1

[26] Yang Liu, Qingchao Chen, and Andrew Zisserman. Amplifying key cues for human-object-interaction detection. In

European Conference on Computer Vision, pages 248–265. Springer, 2020. 2, 6, 7

[27] Ye Liu, Junsong Yuan, and Chang Wen Chen. Consnet: Learning consistency graph for zero-shot human-object interaction detection. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 4235–4243, 2020. 2

[28] Siyuan Qi, Wenguan Wang, Baoxiong Jia, Jianbing Shen, and Song-Chun Zhu. Learning human-object interactions by graph parsing neural networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 401–417, 2018. 2, 7

[29] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28:91–99, 2015. 2

[30] Masato Tamura, Hiroki Ohashi, and Tomoaki Yoshinaga. Qpic: Query-based pairwise human-object interaction detection with image-wide contextual information. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10410–10419, 2021. 1, 2, 3, 5, 6, 7, 8

[31] Oytun Ulutan, ASM Iftekhar, and Bangalore S Manjunath. Vsgnet: Spatial attention network for detecting human object interactions using graph convolutions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13617–13626, 2020. 2, 6

[32] Bo Wan, Desen Zhou, Yongfei Liu, Rongjie Li, and Xuming He. Pose-aware multi-level feature network for human object interaction detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9469–9478, 2019. 2, 6, 7

[33] Hai Wang, Wei-shi Zheng, and Ling Yingbiao. Contextual heterogeneous graph network for human-object interaction detection. In *European Conference on Computer Vision*, pages 248–264. Springer, 2020. 2, 6

[34] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5694–5702, 2019. 7

[35] Tiancai Wang, Tong Yang, Martin Danelljan, Fahad Shahbaz Khan, Xiangyu Zhang, and Jian Sun. Learning human-object interaction detection using interaction points. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4116–4125, 2020. 2, 6, 7

[36] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019. 8

[37] Tete Xiao, Quanfu Fan, Dan Gutfreund, Mathew Monfort, Aude Oliva, and Bolei Zhou. Reasoning about human-object interactions through dual attention networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3919–3928, 2019. 4

[38] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *Thirty-second AAAI conference on artificial intelligence*, 2018. 1, 6

[39] Aixi Zhang, Yue Liao, Si Liu, Miao Lu, Yongliang Wang, Chen Gao, and Xiaobo Li. Mining the benefits of two-stage and one-stage hoi detection. *Advances in Neural Information Processing Systems*, 34, 2021. 2

[40] Frederic Z Zhang, Dylan Campbell, and Stephen Gould. Spatially conditioned graphs for detecting human-object interactions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13319–13327, 2021. 2, 6, 7

[41] Xubin Zhong, Changxing Ding, Xian Qu, and Dacheng Tao. Polysemy deciphering network for human-object interaction detection. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16*, pages 69–85. Springer, 2020. 6

[42] Xubin Zhong, Xian Qu, Changxing Ding, and Dacheng Tao. Glance and gaze: Inferring action-aware points for one-stage human-object interaction detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13234–13243, 2021. 6

[43] Penghao Zhou and Mingmin Chi. Relation parsing neural network for human-object interaction detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 843–851, 2019. 7

[44] 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. 1, 2, 5, 6, 7, 8