

Article

Pairwise CNN-Transformer Features for Human–Object Interaction Detection

Hutuo Quan ^{1,2} , Huicheng Lai ^{1,2,*} , Guxue Gao ^{1,2} , Jun Ma ^{1,2}, Junkai Li ^{1,2} and Dongji Chen ^{1,2}

¹ College of Computer Science and Technology, Xinjiang University, Urumqi 830017, China; quanhutuo@stu.xju.edu.cn (H.Q.); gaoyangshuang123@stu.xju.edu.cn (G.G.); majun@stu.xju.edu.cn (J.M.); lijunkai@stu.xju.edu.cn (J.L.); 107552101245@stu.xju.edu.cn (D.C.)

² Xinjiang Key Laboratory of Signal Detection and Processing, Xinjiang University, Urumqi 830017, China

* Correspondence: lai@xju.edu.cn

Abstract: Human–object interaction (HOI) detection aims to localize and recognize the relationship between humans and objects, which helps computers understand high-level semantics. In HOI detection, two-stage and one-stage methods have distinct advantages and disadvantages. The two-stage methods can obtain high-quality human–object pair features based on object detection but lack contextual information. The one-stage transformer-based methods can model good global features but cannot benefit from object detection. The ideal model should have the advantages of both methods. Therefore, we propose the Pairwise Convolutional neural network (CNN)-Transformer (PCT), a simple and effective two-stage method. The model both fully utilizes the object detector and has rich contextual information. Specifically, we obtain pairwise CNN features from the CNN backbone. These features are fused with pairwise transformer features to enhance the pairwise representations. The enhanced representations are superior to using CNN and transformer features individually. In addition, the global features of the transformer provide valuable contextual cues. We fairly compare the performance of pairwise CNN and pairwise transformer features in HOI detection. The experimental results show that the previously neglected CNN features still have a significant edge. Compared to state-of-the-art methods, our model achieves competitive results on the HICO-DET and V-COCO datasets.

Keywords: computer vision; human–object interaction detection; convolutional neural network; transformer; feature fusion



Citation: Quan, H.; Lai, H.; Gao, G.; Ma, J.; Li, J.; Chen, D. Pairwise CNN-Transformer Features for Human–Object Interaction Detection. *Entropy* **2024**, *26*, 205. <https://doi.org/10.3390/e26030205>

Academic Editor: José F. F. Mendes

Received: 23 November 2023

Revised: 1 February 2024

Accepted: 20 February 2024

Published: 27 February 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Human–object interaction (HOI) detection aims to recognize interactions between humans and objects in a scene. This task requires the precise localization of human–object pairs and the identification of associated actions. HOI detection can facilitate the development of many tasks, such as anomaly detection [1], image understanding [2], and action recognition [3]. HOI detection methods [4] can be categorized into two-stage methods [5–7], traditional one-stage methods [8–10], and end-to-end one-stage methods [11,12]. Two-stage methods leverage an object detector to obtain confidence scores for humans and objects, subsequently filtering out low scores while retaining high-quality instances. However, these methods face challenges in capturing robust contextual cues. Traditional one-stage methods improve inference speed by parallelizing object detection and interaction prediction. However, matching human–object pairs and interactions requires complex strategies. Inspired by DETection TRansformer (DETR) [13], the end-to-end one-stage methods maintain a concise architecture and avoid the matching process of human–object pairs and interactions. These methods leverage global features from the transformer [14], providing the model with rich contextual information. However, they cannot acquire prior knowledge from object detection. Based on the analysis above, our goal is to design a unified model that combines the advan-

tages of both methods. The model can take advantage of the object detector and also has rich contextual information.

Determining how to use the object detector effectively. We note that feature pyramid network (FPN) [15] and HRNet [16] successfully enhance the representation of features by fusing low-resolution features with high-resolution features. Inspired by this, we explore using an object detector to extract multiple types of features. This aims to enhance the features representation.

Current one-stage methods [11,17] outperform two-stage methods [18,19] across the board, so many studies ignore the potential of two-stage methods. Recently, UPT [20] analyzes why one-stage methods are superior to two-stage methods. It is mainly due to using more efficient object detectors and transformer representation capabilities. The two-stage method SCG [21] achieved a significant performance improvement by replacing Faster R-CNN [22] with DETR [13] (object detector). Its performance is close to that of the one-stage method QPIC [11]. Inspired by UPT, our model selects DETR as the object detector. Since DETR inspired a series of one-stage transformer-based methods, these methods use both the convolutional neural network (CNN) backbone and the transformer architecture [14]. However, existing research focuses on using the transformer features to predict HOIs [11,17,23–25] while ignoring the CNN features. Thus, the intuitive scheme is to fuse CNN and transformer features.

Determining how to obtain contextual cues. The interaction head of the two-stage methods [20,26] uses a transformer encoder structure, where the input is only the human–object pair features. The one-stage methods [11,17] use a transformer decoder structure, where the inputs are global features and learnable queries. In two-stage methods, the simple concatenation of human and object features lacks sufficient contextual information. However, the global features from the transformer have proven effective in one-stage methods. Therefore, we propose introducing these global features to provide more explicit contextual cues. In addition, introducing global features can be used to visualize the attention of the model, facilitating future research.

To achieve the above goals, we propose the Pairwise CNN-Transformer (PCT), a simple and intuitive two-stage detector that utilizes pairwise CNN and pairwise transformer features. Figure 1 illustrates our PCT pipeline:

1. We extract features of human and object instances from both the CNN backbone and the transformer architecture.
2. The instance features are fused and paired to form enhanced human–object pair features.
3. The enhanced features are fed into the interaction head with global features to predict actions.

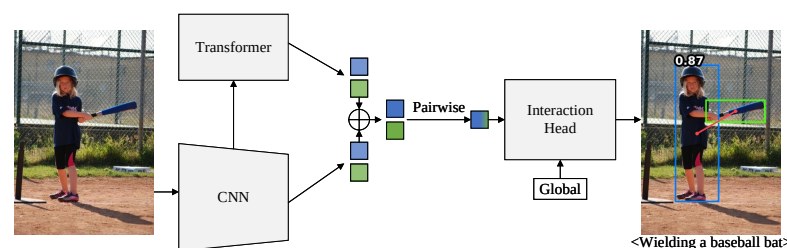


Figure 1. Our PCT pipeline. PCT is a two-stage transformer-based HOI detector. We use blue and green to represent humans and objects, respectively.

Determining why previous work did not compare transformer features and CNN features. Since one-stage transformer-based methods [11,17] use a set of queries to predict HOIs, the number of learnable action queries is fixed. Two-stage methods [5,6,20] generate human–object pairs by pairwise matching. The number of human–object pairs is variable. Different selection strategies for human–object pairs make it challenging to compare the performance of CNN and transformer features fairly. Although UPT [20] has the conditions

to compare them, it does not investigate this issue. To fairly compare their performance in HOI detection, we construct three variant models of UPT. Experimental results show that CNN features outperform transformer features in three sets of comparative experiments.

In summary, our contributions are as follows:

1. We propose fusing the CNN and transformer features of the instances to enhance the feature representations. These enhanced representations effectively improve the precision of HOI detection while introducing little computational cost.
2. We propose utilizing global features from the transformer to enrich contextual information. Through cross-attention, human–object pair features can aggregate valuable information from these global features.
3. We fairly compare the performance of pairwise CNN and pairwise transformer features in HOI detection. The results show that CNN features outperform transformer features.
4. Our model achieves competitive performance on two commonly used HOI datasets, HICO-DET and V-COCO.

We propose a simple and efficient two-stage method PCT. Compared with the recent two-stage method UPT, our method performs better in terms of accuracy, trainable parameters, and inference time. In addition, two-stage methods have a shorter training time and smaller training memory compared to one-stage methods. The code is available at <https://github.com/hutuo1213/PCT> (accessed on 25 January 2024).

2. Related Work

2.1. Object Detection

The object detection task requires locating and identifying objects. This task is the cornerstone of understanding high-level semantics. In recent years, object detection has rapidly evolved from two-stage methods, such as Faster R-CNN [22], to end-to-end methods, such as DETR [13]. Traditional HOI detection methods acquire accurate human and object instances through object detection. Both object detection and HOI detection involve localization and recognition tasks, which makes it possible to migrate object detection strategies to HOI detection. In summary, object detection continues to drive the development of HOI detection. Our method employs a more advanced DETR as the object detector, which consists of a CNN backbone and a transformer network. This method simplifies the detection process by transforming object detection into a set prediction problem.

2.2. Features Fusion

Effective feature fusion enhances the expressive power of features. FPN [15] and HRNet [16] fuse low-resolution features with high-resolution features to enhance features. MSFIN [27] obtains rich information by fusing features from different scales. Kansizoglou et al. [28] explore training strategies for multi-modal fusion. We have observed that explicit methods for features fusion have not been proposed in HOI detection. Therefore, we explore how to fuse features to enhance their representations.

2.3. Human–Object Interaction Detection

HOI detection can be divided into two-stage and one-stage methods. Two-stage methods first perform object detection and then interaction prediction, while one-stage methods simultaneously perform object detection and interaction prediction. HO-RCNN [5], typically a two-stage CNN-based method, proposes using a multi-stream architecture to apply spatial information. The network uses an object detector to obtain spatial information from human–object pairs to construct spatial features. Subsequent studies have constructed different feature priors based on object detection, including visual features of human–object pairs [7], word embeddings of objects [24,29,30], human pose features [19], and body part features [31,32]. ICAN [6] uses an attention module to aggregate HOI cues selectively. MLCNet [19] combines almost all available feature priors to predict HOI. In addition, graph neural networks have been used to reason about relationships between humans and

objects [18,30]. One-stage CNN-based HOI methods [8–10] directly predict interactions, increasing the speed of model inference. However, they must match human–object pairs and interactions, resulting in a more complex design.

Inspired by DETR [13], one-stage transformer-based methods are beginning to grow in popularity. Traditional two-stage methods use pairwise matching of humans and objects to predict HOIs. One-stage transformer-based methods directly predict HOIs using a set of queries. QPIC [11] modifies DETR for HOI detection tasks. HOTR [12] uses interaction representations to infer relevant human–object pairs. DOQ [24] performs knowledge distillation based on QPIC. CDN [17] avoids the challenges of multi-task learning through cascading object detection and action classification modules. GEN-VLKT [25] employs the powerful language–image model CLIP [33] to guide feature learning. These one-stage methods utilize the transformer’s global features to obtain good performance in HOI detection. So, the global features contain rich interaction cues.

In our work, we combine the advantages of the two-stage and one-stage methods. Like UPT [20] and STIP [34], our method is a two-stage transformer-based method. However, our research focuses on improving the expressiveness of features, while UPT and STIP are more focused on model design.

3. Method

The HOI detection task is to accurately locate humans and objects in a scene and identify action categories among them. The principle of our method is based on the Information Gain (IG) in information theory. Specifically:

$$\text{Gain}(P, F) = H(P) - H(P|F) \quad (1)$$

where P is the prediction set of the model, and F is the interaction features. $\text{Gain}(P, F)$ denotes the IG corresponding to the performance in HOI detection. $H(P)$ and $H(P|F)$ denote information and conditional entropy, respectively. $H(\cdot)$ corresponds to the network model. We can reduce the prediction uncertainty ($H(P|F)$ decreases) by enhancing the expression of the interaction feature F . Ultimately, a larger IG is obtained. Section 3.1 introduces our PCT framework. Sections 3.2 and 3.3 describe the PCT’s two critical components. Section 3.4 presents the training and inference details of the PCT.

3.1. Overview

Figure 2 shows our PCT framework. The weights of DETR are frozen throughout the entire training process. In the DETR [13] part, the image $i \in \mathbb{R}^{3 \times H \times W}$ is input to the CNN backbone to generate feature maps $z \in \mathbb{R}^{2048 \times \frac{H}{32} \times \frac{W}{32}}$. After adding positional encoding, the feature map z performs self-attention (transformer encoder) to model global features $G \in \mathbb{R}^{(\frac{H}{32} \times \frac{W}{32}) \times 256}$. Subsequently, the global features G and learnable object queries $q \in \mathbb{R}^{100 \times 256}$ perform cross-attention (transformer decoder) to generate human and object instances $f \in \mathbb{R}^{100 \times 256}$. These instances are decoded into object categories and bounding boxes through a multilayer perceptron (MLP). After filtering out instances with low confidence, the retained bounding boxes are used to extract human and object instances from the CNN backbone network. Then, human and object instances are pairwise matched to generate pairwise CNN features and pairwise transformer features. Add these CNN and transformer features to obtain human–object pair features. In the interaction head part, the human–object pairs and global features perform cross-attention (transformer decoder layer) to aggregate contextual information. Next, these human–object pair features perform self-attention (transformer encoder layer) to refine them [20]. The refined features are decoded into action categories by an MLP.

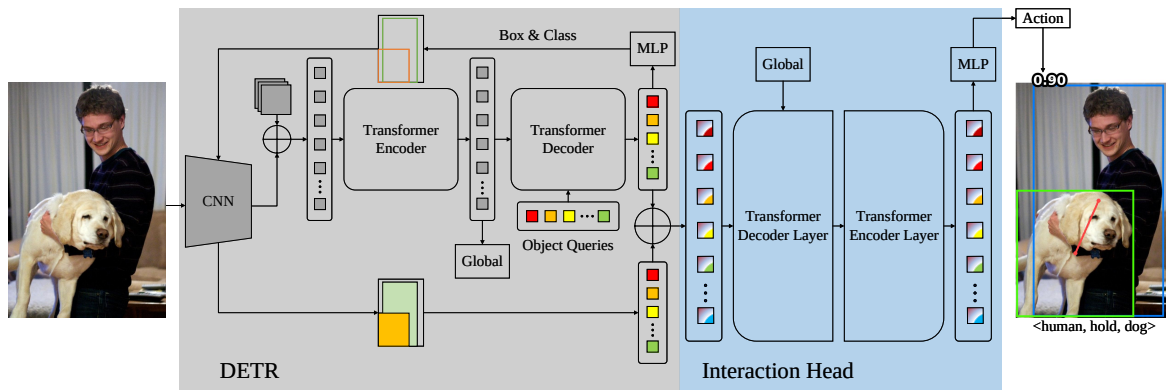


Figure 2. Our PCT framework comprises an object detector (DETR) and an interaction head. First, DETR detects humans and objects in the image, excluding instances with low confidence. Next, the retained humans and objects form human–object pairs through pairwise matching. Finally, the features of these human–object pairs and the global features are fed into the interaction head to predict action categories.

3.2. CNN Features for Human–Object Pairs

Despite the success of the transformer in DETR, it has certain limitations in capturing sequential sequences. CNN are more suitable for capturing the local information than transformers. Therefore, we fuse CNN and transformer features to improve the expressiveness of features.

First, we discussed how DETR works [35]. Figure 3a shows the original image. Figure 3b shows the encoder self-attention at a reference point. We observe that the encoder can separate different object instances. Figure 3c is the decoder attention of an object query. The decoder focuses on features at the edges of objects, such as legs and heads, to better distinguish different objects.

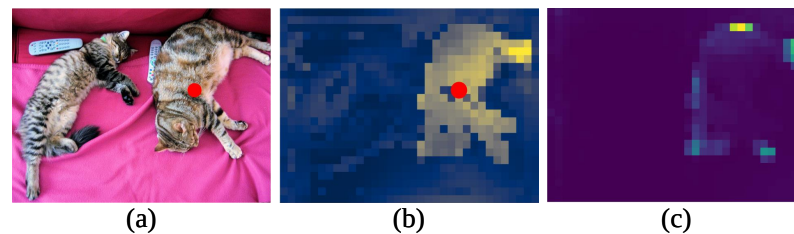


Figure 3. (a) Image with reference point. (b) Encoder self-attention for a reference point. (c) Decoder attention for an object query [35].

Then, we consider how to achieve this goal. Object queries can activate the corresponding object regions in the image. Therefore, the final representations contain object features (transformer features). In addition, the CNN features of objects can be obtained from the region of interest (ROI) in the CNN backbone. Therefore, we can easily align and fuse the CNN and transformer features of instances.

Specifically, according to the detection results of DETR, we first use non-maximum suppression and set a threshold to retain high-score instances. These instances build a set $\{d_i\}_{i=1}^n$, where $d_i = (\mathbf{b}_i, s_i, c_i, f_i)$. $\mathbf{b}_i \in \mathbb{R}^4$ is the object bounding box, $s_i \in [0, 1]$ is the confidence score, $c_i \in \mathcal{K}$ is the object class, $\mathcal{K}$ is the set of object categories, and $f_i \in \mathbb{R}^{1 \times 256}$ is a human or object instance. The transformer features set $\{f_i\}_{i=1}^n$ comprises n_1 humans ($f_{T-h} \in \mathbb{R}^{n_1 \times 256}$) and n_2 objects ($f_{T-o} \in \mathbb{R}^{n_2 \times 256}$). Next, we map the bounding boxes $\mathbf{b}$ to the last layer of the ResNet [36] backbone to obtain ROI. Subsequently, global average pooling is applied to these regions to generate CNN features. Finally, the dimensions of the CNN and transformer features are aligned. Specifically:

$$f_{Ins} = GAP(ROI(z, \mathbf{b})) \quad (2)$$

$$f_{C-h} = \text{Relu}(\text{LN}_1(W_1(f_{Ins-h}))) \quad f_{C-o} = \text{Relu}(\text{LN}_2(W_2(f_{Ins-o}))) \quad (3)$$

where $z \in \mathbb{R}^{2048 \times \frac{H}{32} \times \frac{W}{32}}$ denotes the last layer feature map of the ResNet backbone. $\mathbf{b} \in \mathbb{R}^{n \times 4}$ denotes the object bounding box, ROI denotes the region of interest operation, and GAP denotes the global average pooling operation. $f_{Ins} \in \mathbb{R}^{n \times 2048}$ represents instance features of both humans and objects. $W_1, W_2 \in \mathbb{R}^{2048 \times 256}$ denotes the linear layer, LN_1, LN_2 denotes the layer normalization, and Relu denotes the Relu function. $f_{C-h} \in \mathbb{R}^{n1 \times 256}$ and $f_{C-o} \in \mathbb{R}^{n2 \times 256}$ denote the CNN features of the human and object, respectively. Next, as shown in Figure 4, We pairwise match human and object instances by concatenation. $n1$ humans and $n2$ objects will generate $m = n1 \times (n1 + n2 - 1)$ human–object pairs. Then, the pairwise CNN features and pairwise transformer features are added element-wise. Specifically:

$$Z = (f_{T-h} \oplus f_{T-o}) + (f_{C-h} \oplus f_{C-o}) \quad (4)$$

where $\oplus$ denotes concatenation, $+$ denotes element-wise addition. $Z \in \mathbb{R}^{m \times 512}$ denotes the human–object pair features, where m denotes the number of pairs.

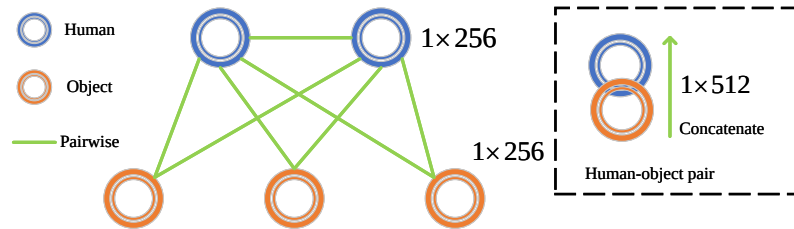


Figure 4. The process of pairwise matching between humans and objects.

3.3. Interaction Head

We argue that simply connecting humans and objects does not provide contextual clues. Given that previous studies often use global features to provide interaction cues, we introduce global features in the interaction head. In Section 4.6, we visualize the decoder attention of the interaction head. Human–object pairs are effectively localized and highlighted when global features are applied. Specifically, the global features are extracted from the transformer encoder of DETR [13]. We use a standard transformer decoder layer that performs cross-attention on human–object pair and global features, as follows:

$$G = \text{Relu}(\text{LN}_3(W_3(G))) \quad (5)$$

$$\text{output} = \text{Dec}(Q = Z, K = V = G) \quad (6)$$

where $W_3 \in \mathbb{R}^{256 \times 512}$ denotes the linear layer. The global features $G \in \mathbb{R}^{(\frac{H}{32} \times \frac{W}{32}) \times 256}$ are updated to $G \in \mathbb{R}^{(\frac{H}{32} \times \frac{W}{32}) \times 512}$. Dec denotes the transformer decoder layer, and $\text{output} \in \mathbb{R}^{m \times 512}$ denotes output features.

Inspired by UPT [20], we used a transformer encoder layer to refine the features of human–object pairs. Specifically:

$$\text{output} = \text{Enc}(\text{output}) \quad (7)$$

where Enc denotes the transformer encoder layer.

Finally, we decode the output into action classification scores. Specifically:

$$\tilde{\mathbf{s}}_k = W(\text{output}) \quad (8)$$

$\tilde{\mathbf{s}}_k$ denotes the unnormalized action score. k is the k -th possible action of the human–object pair. W denotes the linear layer. For HICO-DET [5], we use $W \in \mathbb{R}^{512 \times 117}$; for V-COCO [37], we use $W \in \mathbb{R}^{512 \times 24}$.

3.4. Training and Inference

3.4.1. Training

Following UPT, we use focal loss [38] with a sigmoid layer to alleviate the imbalance between positive and negative samples and maintain numerical stability [20]. The specific formula is as follows:

$$\hat{y} = \mathbf{s}_h \cdot \mathbf{s}_o \cdot \sigma(\tilde{\mathbf{s}}_k) \quad (9)$$

$$\text{FL}(\hat{y}, y) = \begin{cases} -\beta(1 - \hat{y})^\gamma \log(\hat{y}), & y = 1 \\ -(1 - \beta) \hat{y}^\gamma \log(1 - \hat{y}), & y = 0 \end{cases} \quad (10)$$

where $\mathbf{s}_h$ and $\mathbf{s}_o$ are the confidence scores of humans and objects, respectively. σ is the sigmoid function, $\sigma(\tilde{\mathbf{s}}_k) \in [0, 1]$ is the normalized action score, and $\hat{y} \in [0, 1]$ is the final action score. $y \in \{0, 1\}$ is the binary ground truth label. $\beta \in [0, 1]$ and $\gamma \in \mathbb{R}_+$ are hyperparameters.

3.4.2. Inference

Based on previous research [5], we excluded invalid combinations of actions and objects, such as holding an airplane. After SCG [21], we implemented an instance suppression strategy [20,31] (as shown in Figure 5) to avoid excessive object scores dominating the final HOI scores. Specifically, the object scores are reduced by adding a power of 2.8, and the final HOI scores $\mathbf{s}_k$ are calculated as follows:

$$\mathbf{s}_k = (\mathbf{s}_h)^\lambda \cdot (\mathbf{s}_o)^\lambda \cdot \sigma(\tilde{\mathbf{s}}_k) \quad (11)$$

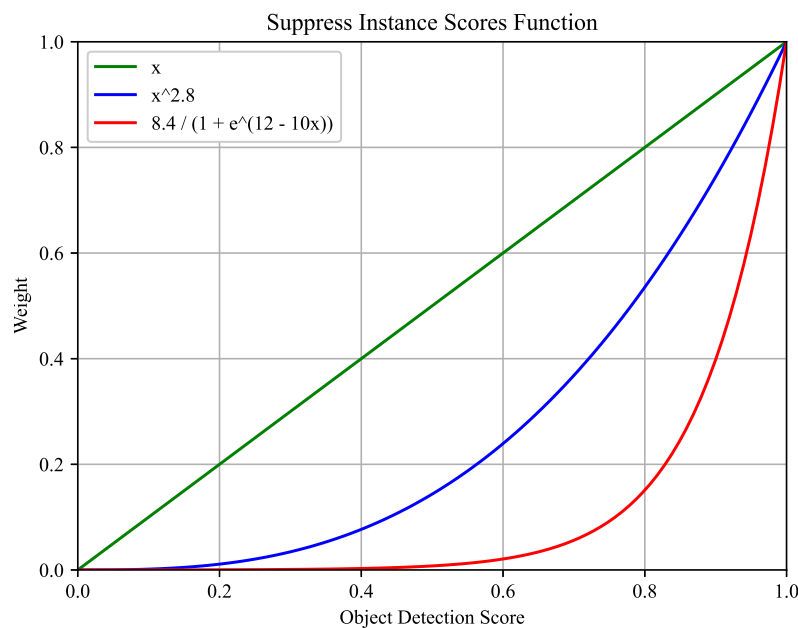


Figure 5. The green curve indicates no instance suppression fraction. The red and blue curves are the instance suppression strategies of TIN [31] and SCG [21], respectively. We follow the SCG.

4. Experiments

Section 4.1 details our experiment settings. Section 4.2 compares the results of our method with the state-of-the-art methods. Section 4.3 presents our model's ablation study and comparative study. Section 4.4 demonstrates the costing study. Section 4.5 compares the expressive power of CNN features and transformer features. Section 4.6 presents qualitative results and limitations. If not specified, the models in this section use the ResNet-50 backbone and report their performance on the HICO-DET [5] dataset.

4.1. Experiment Settings

4.1.1. Datasets

The HICO-DET dataset contains 37,633 valid training images and 9546 valid testing images, covering 80 objects, 117 actions, and 600 interactions. The V-COCO [37] dataset contains 4969 valid training images and 4532 valid testing images, covering 29 actions. Following UPT [20], we report results for 24 actions in the V-COCO dataset.

4.1.2. Evaluation Metrics

We report the mean Average Precision (mAP) on the HICO-DET and V-COCO datasets. A prediction is considered a true positive if the intersection over union (IOU) of the human and object boxes with the ground truth boxes is higher than 0.5, and the HOI class is correct. For HICO-DET, we evaluate mAP on default and known object settings. Each setting contains 3 HOI sets: full (600 HOIs), rare (138 HOIs, less than 10 training instances), and non-rare (462 HOIs, 10 or more training instances). For V-COCO, we evaluate the mAP of Scenario 1 and Scenario 2. For the cases without role annotations, Scenario 1 requires the role prediction to be null, while Scenario 2 ignores the role prediction.

4.1.3. Implementation Details

For HICO-DET, we use the DETR model [20] fine-tuned on HICO-DET. For V-COCO, we use the DETR model [20] trained from scratch on MS COCO [39]. Our data augmentation techniques include random cropping, image scaling, and color dithering [11,13]. We collect instances with scores greater than 0.2 and keep a maximum of 15 humans and 15 objects. The hyperparameters of focal loss are the same as UPT [20]. Our model is trained for 20 epochs using the AdamW [40] optimizer. The initial learning rate is 10^{-4} , and the learning rate is reduced to 10^{-5} at the 10th epoch. The training uses a NVIDIA Tesla V100 device with a batch size of 16.

4.2. Comparisons with State-of-the-Art Methods

Tables 1 and 2 compares the performance of our PCT with other methods. Although we do not conduct studies on aspects such as data augmentation and loss functions, such studies may complement ours. Our PCT with ResNet-50 achieves 33.63 mAP on the HICO-DET dataset, comparable to state-of-the-art methods. Notably, the PCT outperforms some models [17,20,41] with more powerful backbone networks. Our PCT achieves excellent performance using only visual features, indicating that our model is efficient. On the V-COCO dataset, our PCT with ResNet-101 significantly improves (about 2 mAP) over ResNet-50. In Scenario 2, the former achieves 67.1 mAP performance, comparable to state-of-the-art methods.

Table 1. Performance (mAP $\times$ 100) comparison on the HICO-DET dataset. The highest result for each setting is shown in bold.

Method	Backbone	Default Setting			Known Objects Setting		
		Full	Rare	Non-Rare	Full	Rare	Non-Rare
CNN-based methods							
HO-RCNN [5]	CaffeNet	7.81	5.37	8.54	10.41	8.94	10.85
InteractNet [7]	Res50-FPN	9.94	7.16	10.77	-	-	-
ICAN [6]	Res50	14.84	10.45	16.15	16.26	11.33	17.73
Xia et al. [42]	Res50	15.85	13.12	16.65	-	-	-
UnionDet [10]	Res50-FPN	17.58	11.72	19.33	19.76	14.68	21.27
VSGNet [18]	Res152	19.80	16.05	20.91	-	-	-
TIN [31]	Res50-FPN	20.93	18.95	21.32	23.02	20.96	23.42
PPDM [8]	Hourglass104	21.94	13.97	24.32	24.81	17.09	27.12
DRG [29]	Res50-FPN	24.53	19.47	26.04	27.98	23.11	29.43
SKGHOI [43]	Res50-FPN	26.95	21.28	28.56	-	-	-
SCG [21]	Res50-FPN	29.26	24.61	30.65	32.87	27.89	34.35

Table 1. Cont.

Method	Backbone	Default Setting			Known Objects Setting		
		Full	Rare	Non-Rare	Full	Rare	Non-Rare
Transformer-based methods							
HOTR [12]	Res50	25.10	17.34	27.42	-	-	-
HOI-Trans [44]	Res101	26.61	19.15	28.84	29.13	20.98	31.57
AS-Net [23]	Res50	28.87	24.25	30.25	31.74	27.07	33.14
QPIC [11]	Res101	29.90	23.92	31.69	32.38	26.06	34.27
PhraseHOI [45]	Res101	30.03	23.48	31.99	33.74	27.35	35.64
MSTR [46]	Res50	31.17	25.31	32.92	34.02	28.83	35.57
CDN [17]	Res101	32.07	27.19	33.53	34.79	29.48	36.38
UPT [20]	Res101	32.31	28.55	33.44	35.65	31.60	36.86
Iwin [41]	Res101-FPN	32.79	27.84	35.40	35.84	28.74	36.09
PR-Net [47]	Res101	32.86	28.03	34.30	-	-	-
Ours (PCT)	Res50	33.63	28.73	35.10	36.96	31.47	38.60
Ours (PCT)	Res101	33.79	29.70	35.00	37.16	32.63	38.52

Table 2. Performance (mAP $\times 100$) comparison on the V-COCO dataset. The highest result for each setting is shown in bold.

Method	Backbone	AP ^{S1} _{role}	AP ^{S2} _{role}
CNN-based methods			
InteractNet [7]	Res50-FPN	40.0	-
ICAN [6]	Res50	45.3	-
Xia et al. [42]	Res50	47.2	-
UnionDet [10]	Res50-FPN	47.5	56.2
VSGNet [18]	Res152	51.8	57.0
TIN [31]	Res50-FPN	49.1	-
DRG [29]	Res50-FPN	51.0	-
SCG [21]	Res50-FPN	54.2	60.9
Transformer-based methods			
HOTR [12]	Res50	55.2	64.4
HOI-Trans [44]	Res101	52.9	-
AS-Net [23]	Res50	53.9	-
QPIC [11]	Res101	58.3	60.7
MSTR [46]	Res50	62.0	65.2
CDN [17]	Res101	63.9	65.9
UPT [20]	Res101	60.7	66.2
Iwin [41]	Res101-FPN	60.9	-
PR-Net [47]	Res101	62.9	64.2
Ours (PCT)	Res50	59.4	65.0
Ours (PCT)	Res101	61.4	67.1

We observed that the performance of PCT using ResNet-50 and PCT using ResNet-101 is similar on the HICO-DET dataset. Typically, features extracted by ResNet-101 are superior to those of ResNet-50. To analyze this phenomenon, Table 3 demonstrates the impact of backbone networks and instance suppression strategies on performance.

Without performing instance suppression ($\lambda = 1$), PCT with ResNet-101 outperforms PCT with ResNet-50 by 0.38 mAP. As λ gradually increases to 2.8, the performance gap between these two models decreases to 0.16 mAP. Additionally, the instance suppression strategy proposed by TIN [31] also reduces the performance gap between these two models. Experimental results show that the instance suppression strategies narrow the performance gap between these two models.

Table 3. Results of implementing the instance suppression strategies.

Backbone	Instance Suppression Strategies			
	$\lambda = 1.0$	$\lambda = 1.9$	$\lambda = 2.8$	TIN [31]
Res50	32.10	33.44	33.63	32.92
Res101	32.48	33.63	33.79	33.11
Δ	+0.38	+0.19	+0.16	+0.19

4.3. Ablation and Contrast Studies

4.3.1. Ablation Study

We show the results of the ablation study in Table 4. The interaction head of the baseline model is a transformer encoder layer. Compared to the baseline, adding a transformer decoder layer improves about 2 mAP, adding the CNN features of human–object pairs improves about 3.3 mAP, and the complete PCT model improves about 4 mAP. In addition, removing the encoder layer from the complete model degrades about 0.8 mAP.

Table 4. Ablation studies of our PCT components. The interaction head of the baseline model uses a standard encoder layer.

Model	CNN Features	Decoder	Full	Rare	Non-Rare
Baseline	-	-	29.48	24.46	30.97
	-	✓	31.50	26.98	32.84
	✓	-	32.80	27.89	34.26
Ours	✓	✓	33.63	28.73	35.10

Following the UPT [20], we show the effect of model components on the interaction scores. Since we freeze the object detector, different interaction heads will use the same object detection result. We divide the negative samples according to the baseline model. An interaction score less than 0.05 is defined as an easy negative sample, otherwise a hard negative sample. In Table 5, we explore the impact of each component on positive and negative samples by controlling variables.

Table 5. Change in average interaction score when a component is added to the reference network. According to the prediction results of the baseline model, the samples are categorized as positive, easy negatives, and hard negatives. The number of samples in each category is in parentheses.

Reference	Δ Submodules	Δ Pos. (25,342)	Δ Easy Neg. (4,373,390)	Δ Hard Neg. (46,172)
PCT w/o encoder layer	+ encoder layer	+0.0064	−0.0000	−0.0025
PCT w/o decoder layer	+ decoder layer	+0.0127	−0.0000	+0.0001
PCT w/o CNN features	+ CNN features	+0.0167	−0.0001	−0.0152
Baseline	PCT	+0.0332	−0.0001	−0.0273

Table 5 shows that adding CNN features increases the average score of positive samples by 0.0167 and degrades the average score of hard negative samples by 0.0152, widening the gap between positive and negative sample scores. Adding the transformer decoder layer improves the average score of positive samples by 0.0127, but hard negative samples have almost no effect. Therefore, both of our proposed components are valid. Finally, our complete model improves the average score of positive samples by 0.0332 and reduces the average score of hard negative samples by 0.0273, further enlarging this gap. This indicates that the two components are complementary.

4.3.2. Contrast Study

We show the results of different interaction heads in Table 6. Our interaction head first uses a transformer decoder layer and then a transformer encoder layer (D.E.+E.N., 33.63 mAP). The interaction head (E.N.+D.E., 33.41 mAP) with the opposite order of our submodules drops about 0.2 mAP in performance than ours. We speculate that the decoder layer needs to perform attention and generate HOI representations, which is challenging to balance. In addition, the design is inflexible due to the long global feature migration distance. The interaction head composed of double transformer encoder layers (E.N.+E.N., 33.15 mAP) is about 0.5 mAP lower than ours, which indicates that global features provide effective HOI clues. Although the interaction head consisting of double transformer decoder layers (D.E.+D.E., 33.63 mAP) covers ours, the performance does not improve further.

Table 6. Comparison of results for different interaction heads. The acronym E.N. stands for encoder layer, and the acronym D.E. stands for decoder layer. The order of the acronyms indicates the order of the submodules.

Interaction Head	Full	Rare	Non-Rare
Only E.N.	32.80	27.89	34.26
Only D.E.	32.84	28.48	34.14
E.N.+E.N.	33.15	28.03	34.68
E.N.+D.E.	33.41	28.64	34.84
D.E.+D.E.	33.63	28.46	35.18
D.E.+E.N. (ours)	33.63	28.73	35.10

4.4. Computational Costing Study

Our network structure is straightforward, but the feature extraction method may increase the model's complexity. To clarify the complexity of our model, Table 7 compares the computational cost of our method with the classical one-stage and two-stage methods. In addition, Table 8 shows the computational cost of our practical components. All experiments were performed on a NVIDIA GeForce GTX 1080 with an average input image size of [3, 887, 1055].

Table 7. Computational cost comparison. DETR is the object detector in our method. The number of trainable parameters is shown in parentheses.

Method	Backbone	Params (M)	FLOPs (G)	FPS
SCG	Res50-FPN	54.16	130.90	11.98
DETR	Res50	41.52	91.73	13.90
	Res101	60.46	163.70	9.61
QPIC	Res50	41.46	91.82	13.78
	Res101	60.40	163.79	9.62
UPT	Res50	54.76 (13.24)	91.91	12.23
	Res101	73.70 (13.24)	163.97	8.84
PCT(ours)	Res50	50.12 (8.60)	92.47	13.02
	Res101	69.06 (8.60)	164.49	9.27

Table 7 shows that our model (13.02 fps) only lags behind the one-stage method QPIC (13.78 fps) in inference speed but outperforms other HOI detectors. Both UPT and our method are two-stage transformer-based methods. Compared to UPT (13.24 M), our model (8.60 M) has fewer trainable parameters, higher FPS, and is only slightly higher than UPT in terms of FLOPs. As shown in Figure 6, although UPT has lower FLOPs than ours, it has more submodules, which affects the inference speed.

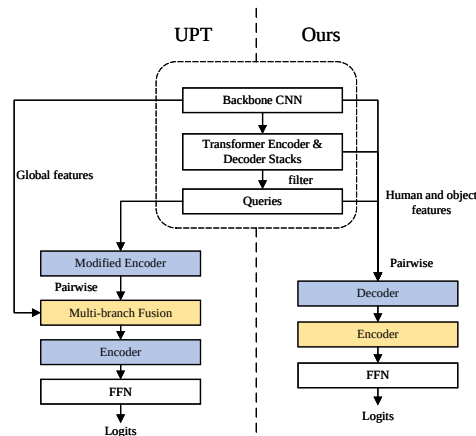


Figure 6. Comparison between UPT and our model.

Table 8 shows that our key components add a small number of parameters and a slight FPS reduction. The main reason for the increased FLOPs is caused by the decoder layer, which performs cross-attention between global features and human–object pair features.

Table 8. Costs of our model components. The Params column shows the number of trainable parameters. Changes in the number of parameters are shown in parentheses. All methods use resnet50 as the backbone network.

Method	Params (M)	FLOPs (G)	FPS
UPT	13.24	91.913	12.23
Only E.N. (Baseline)	3.21 (+0.00)	91.773	13.41
+CNN	4.26 (+1.05)	91.777	13.18
+D.E.	7.55 (+4.34)	92.468	13.21
+all (ours)	8.60 (+5.39)	92.472	13.02

4.5. Comparing CNN and Transformer Features

To fairly compare the performance of CNN and transformer features in HOI detection, we designed three variant models based on UPT. These models follow the structure shown in Figure 7, where the backbone CNN and the cooperative layer (Coop.) can be replaced. The modified encoder (M.E.) adds spatial information compared to the vanilla encoder (E.N.).

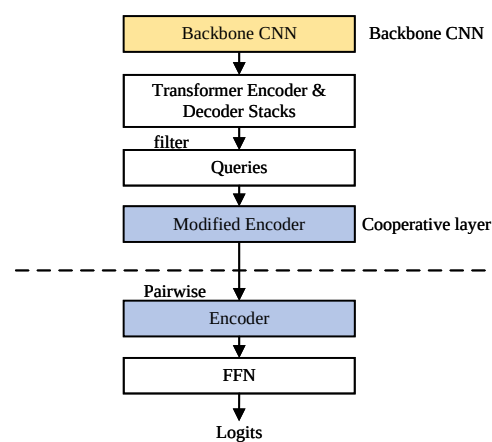


Figure 7. Framework of the UPT variant. The position before pairwise operations is called the cooperative layer.

Table 9 shows that CNN features outperform transformer features in all variant models. When the vanilla transformer encoder layer (E.N.) is used in the cooperative layer, CNN features (31.16 mAP) outperform transformer features (30.39 mAP). We attribute this to the better local detail and stability of CNN features. Notably, when the cooperative layer uses the modified transformer encoder layer (M.E.), CNN features (32.17 mAP) further improve performance, while transformer features (30.22 mAP) perform unfavorably. We attribute this to the modified transformer encoder providing the spatial information, but the transformer features cannot effectively exploit this spatial information. When we replace the backbone network with the more powerful ResNet-101, improvements are achieved using both CNN and transformer features. Moreover, transformers show strong adaptability [20,26] and expressiveness.

Table 9. Comparison of the results of CNN features and transformer features. The acronym E.N. stands for standard encoder layer, and M.E. stands for modified encoder layer. In addition, the cooperative layer does not use the dropout layer.

Feature	Coop.	Full	Rare	Non-Rare
transformer	E.N.	30.39	24.95	32.02
CNN	E.N.	31.16	27.12	32.37
transformer	M.E.	30.22	24.71	31.86
CNN	M.E.	32.17	26.61	33.82
transformer	M.E.&Res101	30.47	26.38	31.69
CNN	M.E.&Res101	32.23	27.65	33.60

4.6. Qualitative Results and Limitations

Figure 8 shows how the PCT model works qualitatively. We visualize attention at the transformer decoder layer in the interaction head. Figure 8c shows attention using only transformer features. We observe that the related human–object pair and its closer human–object pairs are focused, and we speculate that global features provide implicit spatial clues. As for the model paying attention to neighboring human–object pairs, the reason is the use of self-attention between human–object pairs. Figure 8d shows the attention of the complete PCT model. Using the fusion features of CNN and transformer can focus on relevant areas of human–object pairs more fully. We attribute this to the CNN effectively capturing local features and detailed information.

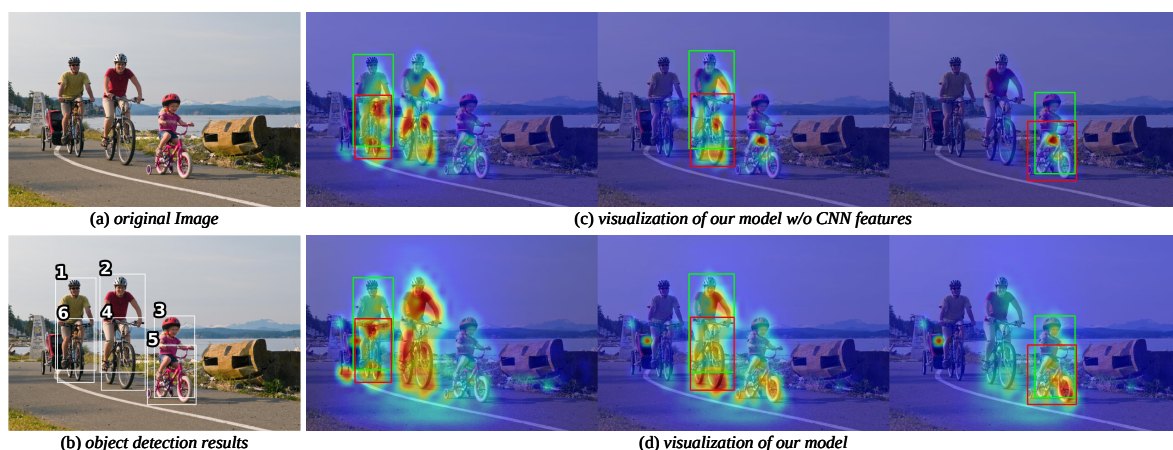


Figure 8. Effect of pairwise CNN features on attention. Please zoom in to view.

Qualitative analysis of HOI detection. Figure 9 demonstrates that our model can accurately identify HOI categories. However, our model does not perform well when the image is strongly occluded (as shown in Figure 9b) because the object score affects the final interaction score. Furthermore, we show several failure examples of the model. In

Figure 10a, the false detection is caused by similar spatial relations with the straddling action. In Figure 10b, false detection occurs due to strong occlusion. In Figure 10c, the interaction score is too low due to fewer training samples. In Figure 10d, not detecting the correct object leads to wrong predictions. In Figure 10e, semantic ambiguity is caused by ambiguous interaction actions. These qualitative analyses provide valuable clues for model optimization.

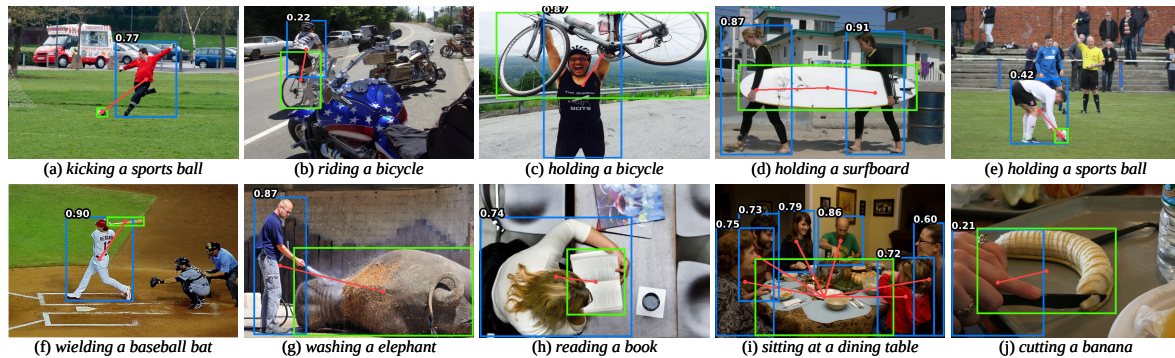


Figure 9. Successful cases. We filter out human–object pairs with scores lower than 0.2. Please zoom in to view.



Figure 10. Failure cases. (a) Typical false detection. (b) Typical missed detection. (c) Low score. (d) Wrong object. (e) Semantic ambiguity.

5. Conclusions

In this paper, we try to combine the advantages of two-stage and one-stage methods. Our PCT extracts CNN features through an object detector to improve feature representation. Meanwhile, it uses the transformer’s global features to provide contextual information. Our method achieves competitive performance with a simple and efficient architecture. Then, we fairly compare the performance of CNN and transformer features in HOI detection. The results show that CNN features are not inferior to transformer features. Our work may advance the study of feature fusion and lightweight networks for HOI detection. In addition, we speculate that the features generated by the object detector are more suitable for object classification than interaction classification. In future work, we plan to adopt a universal visual pre-trained model to provide robust and high-quality features. Our method relies on human–object pair features to predict interactions, so it cannot predict scenarios that do not involve objects (such as running or falling). In addition, incorrect classification of objects by the object detector leads to wrong HOI predictions.

Author Contributions: Conceptualization, H.Q.; methodology, H.Q.; software, H.Q.; validation, D.C.; formal analysis, J.M.; investigation, J.L.; resources, H.L.; data curation, G.G.; writing—original draft preparation, H.Q.; writing—review and editing, H.Q., J.M., J.L. and D.C.; visualization, H.Q.; supervision, G.G.; project administration, H.L.; funding acquisition, H.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Natural Science Foundation of China grant number U1903213.

Institutional Review Board Statement: Not applicable.

Data Availability Statement: All data available upon reasonable request.

Acknowledgments: We acknowledge Zhen Zhang for his advice.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Xiao, Y.; Gao, G.; Wang, L.; Lai, H. Optical flow-aware-based multi-modal fusion network for violence detection. *Entropy* **2022**, *24*, 939. [[CrossRef](#)] [[PubMed](#)]
2. Lv, J.; Hui, T.; Zhi, Y.; Xu, Y. Infrared Image Caption Based on Object-Oriented Attention. *Entropy* **2023**, *25*, 826. [[CrossRef](#)] [[PubMed](#)]
3. Wang, L.; Yao, W.; Chen, C.; Yang, H. Driving behavior recognition algorithm combining attention mechanism and lightweight network. *Entropy* **2022**, *24*, 984. [[CrossRef](#)]
4. Antoun, M.; Asmar, D. Human object interaction detection: Design and survey. *Image Vis. Comput.* **2023**, *130*, 104617. [[CrossRef](#)]
5. Chao, Y.W.; Liu, Y.; Liu, X.; Zeng, H.; Deng, J. Learning to detect human–object interactions. In Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), Lake Tahoe, NV, USA, 12–15 March 2018; pp. 381–389.
6. Gao, C.; Zou, Y.; Huang, J.B. iCAN: Instance-Centric Attention Network for Human–Object Interaction Detection. In Proceedings of the British Machine Vision Conference (BMVC), Newcastle, UK, 3–6 September 2018; pp. 1–13.
7. Gkioxari, G.; Girshick, R.; Dollár, P.; He, K. Detecting and recognizing human–object interactions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2018, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8359–8367.
8. Liao, Y.; Liu, S.; Wang, F.; Chen, Y.; Qian, C.; Feng, J. 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 (CVPR) 2020, Seattle, WA, USA, 14–19 June 2020; pp. 482–490.
9. Wang, T.; Yang, T.; Danelljan, M.; Khan, F.S.; Zhang, X.; Sun, J. Learning human–object interaction detection using interaction points. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) 2020, Seattle, WA, USA, 14–19 June 2020; pp. 4116–4125.
10. Kim, B.; Choi, T.; Kang, J.; Kim, H.J. Uniondet: Union-level detector towards real-time human–object interaction detection. In *Computer Vision—ECCV 2020*; Springer: Cham, Switzerland, 2020; pp. 498–514.
11. Tamura, M.; Ohashi, H.; Yoshinaga, T. 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 2021, Virtual, 19–25 June 2021; pp. 10410–10419.
12. Kim, B.; Lee, J.; Kang, J.; Kim, E.S.; Kim, H.J. Hotr: End-to-end human–object interaction detection with transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, 19–25 June 2021; pp. 74–83.
13. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. In *Computer Vision—ECCV 2020*; Springer: Cham, Switzerland, 2020; pp. 213–229.
14. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Proc. Adv. Neural Inf. Process. Syst.* **2017**, *30*, 1–11.
15. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017, Honolulu, HI, USA, 21–26 July 2017; pp. 2117–2125.
16. Sun, K.; Xiao, B.; Liu, D.; Wang, J. Deep high-resolution representation learning for human pose estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2019, Long Beach, CA, USA, 15–20 June 2019; pp. 5693–5703.
17. Zhang, A.; Liao, Y.; Liu, S.; Lu, M.; Wang, Y.; Gao, C.; Li, X. Mining the benefits of two-stage and one-stage hoi detection. *Proc. Adv. Neural Inf. Process. Syst.* **2021**, *34*, 17209–17220.
18. Ulutan, O.; Iftekhhar, A.; Manjunath, B.S. 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 2020, Seattle, WA, USA, 14–19 June 2020; pp. 13617–13626.
19. Sun, X.; Hu, X.; Ren, T.; Wu, G. Human object interaction detection via multi-level conditioned network. In Proceedings of the 2020 International Conference on Multimedia Retrieval, Dublin, Ireland, 8–11 June 2020; pp. 26–34.
20. Zhang, F.Z.; Campbell, D.; Gould, S. Efficient two-stage detection of human–object interactions with a novel unary-pairwise transformer. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2022, New Orleans, LA, USA, 18–24 June 2022; pp. 20104–20112.
21. Zhang, F.Z.; Campbell, D.; Gould, S. Spatially conditioned graphs for detecting human–object interactions. In Proceedings of the IEEE/CVF International Conference on Computer Vision 2021, Virtual, 11–17 October 2021; pp. 13319–13327.
22. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*; Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R., Eds.; Curran Associates: Red Hook, NY, USA, 2015; Volume 28, pp. 91–99.
23. Chen, M.; Liao, Y.; Liu, S.; Chen, Z.; Wang, F.; Qian, C. Reformulating hoi detection as adaptive set prediction. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2021, Virtual, 19–25 June 2021; pp. 9004–9013.

24. Qu, X.; Ding, C.; Li, X.; Zhong, X.; Tao, D. Distillation using oracle queries for transformer-based human–object interaction detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2022, New Orleans, LA, USA, 18–24 June 2022; pp. 19558–19567.
25. Liao, Y.; Zhang, A.; Lu, M.; Wang, Y.; Li, X.; Liu, S. Gen-vlkt: Simplify association and enhance interaction understanding for hoi detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2022, New Orleans, LA, USA, 18–24 June 2022; pp. 20123–20132.
26. Wang, G.; Guo, Y.; Wong, Y.; Kankanhalli, M. Distance Matters in Human–Object Interaction Detection. In Proceedings of the 30th ACM International Conference on Multimedia 2022, Lisboa, Portugal, 10–14 October 2022; pp. 4546–4554.
27. Liu, Y.; Wang, L.; Cheng, J.; Chen, X. Multiscale feature interactive network for multifocus image fusion. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–16. [CrossRef]
28. Kansizoglou, I.; Bampis, L.; Gasteratos, A. Deep feature space: A geometrical perspective. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *44*, 6823–6838. [CrossRef] [PubMed]
29. Gao, C.; Xu, J.; Zou, Y.; Huang, J.B. Drg: Dual relation graph for human–object interaction detection. In *Computer Vision–ECCV 2020*; Springer: Cham, Switzerland, 2020; pp. 696–712.
30. Liang, Z.; Liu, J.; Guan, Y.; Rojas, J. Visual-semantic graph attention networks for human–object interaction detection. In Proceedings of the 2021 IEEE International Conference on Robotics and Biomimetics (ROBIO), Sanya, China, 27–31 December 2021; pp. 1441–1447.
31. Li, Y.L.; Liu, X.; Wu, X.; Huang, X.; Xu, L.; Lu, C. Transferable Interactiveness Knowledge for Human–Object Interaction Detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 3870–3882. [PubMed]
32. Wu, X.; Li, Y.L.; Liu, X.; Zhang, J.; Wu, Y.; Lu, C. Mining cross-person cues for body-part interactiveness learning in hoi detection. In *Computer Vision–ECCV 2022*; Springer: Cham, Switzerland, 2022; pp. 121–136.
33. Radford, A.; Kim, J.W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. Learning transferable visual models from natural language supervision. In Proceedings of the International Conference on Machine Learning PMLR, New York, USA 18–24 July 2021; pp. 8748–8763.
34. Zhang, Y.; Pan, Y.; Yao, T.; Huang, R.; Mei, T.; Chen, C.W. Exploring structure-aware transformer over interaction proposals for human–object interaction detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2022, New Orleans, LA, USA, 18–24 June 2022; pp. 19548–19557.
35. DETR’s Hands on Colab Notebook. Facebook AI. Available online: <https://github.com/facebookresearch/detr> (accessed on 26 May 2020).
36. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition 2016, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
37. Gupta, S.; Malik, J. Visual Semantic Role Labeling. *arXiv* **2015**, arXiv:1505.04474.
38. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision 2017, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
39. Lin, T.Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C.L. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014*; Springer: Cham, Switzerland, 2014; pp. 740–755.
40. Loshchilov, I.; Hutter, F. Decoupled Weight Decay Regularization. In Proceedings of the International Conference on Learning Representations 2019, New Orleans, LA, USA, 6–9 May 2019.
41. Tu, D.; Min, X.; Duan, H.; Guo, G.; Zhai, G.; Shen, W. Iwin: Human–object interaction detection via transformer with irregular windows. In *European Conference on Computer Vision*; Springer: Cham, Switzerland, 2022; pp. 87–103.
42. Xia, L.; Li, R. Multi-stream neural network fused with local information and global information for HOI detection. *Appl. Intell.* **2020**, *50*, 4495–4505. [CrossRef]
43. Zhu, L.; Lan, Q.; Velasquez, A.; Song, H.; Kamal, A.; Tian, Q.; Niu, S. SKGHOI: Spatial-Semantic Knowledge Graph for Human–Object Interaction Detection. *arXiv* **2023**, arXiv:2303.04253.
44. Zou, C.; Wang, B.; Hu, Y.; Liu, J.; Wu, Q.; Zhao, Y.; Li, B.; Zhang, C.; Zhang, C.; Wei, Y.; 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 2021, Virtual, 19–25 June 2021; pp. 11825–11834.
45. Li, Z.; Zou, C.; Zhao, Y.; Li, B.; Zhong, S. Improving human–object interaction detection via phrase learning and label composition. In Proceedings of the AAAI Conference on Artificial Intelligence 2022, Online, 22 February–1 March 2022; Volume 36, pp. 1509–1517.
46. Kim, B.; Mun, J.; On, K.W.; Shin, M.; Lee, J.; Kim, E.S. Mstr: Multi-scale transformer for end-to-end human–object interaction detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition 2022, New Orleans, LA, USA, 18–24 June 2022; pp. 19578–19587.
47. Peng, H.; Liu, F.; Li, Y.; Huang, B.; Shao, J.; Sang, N.; Gao, C. Parallel Reasoning Network for Human–Object Interaction Detection *arXiv* **2023**, arXiv:2301.03510.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.