Consistency Learning via Decoding Path Augmentation for Transformers in Human Object Interaction Detection

Jihwan Park[1,2] SeungJun Lee[1] Hwan Heo[1] Hyeong Kyu Choi[1] Hyunwoo J. Kim[1,*]
[1]Department of Computer Science and Engineering, Korea University [2]Kakao Brain
{jseven7071, lapal0413, gjghks950, imhgchoi, hyunwoojkim}@korea.ac.kr
{jwan.park}@kakaobrain.com

Abstract

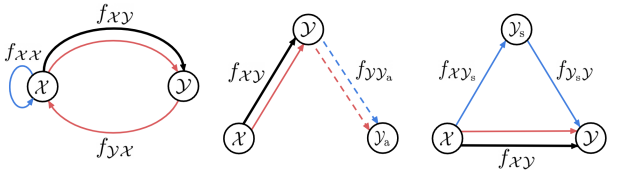
Human-Object Interaction detection is a holistic visual recognition task that entails object detection as well as interaction classification. Previous works of HOI detection has been addressed by the various compositions of subset predictions, e.g., *Image → HO → I, Image → HI → O. Recently, transformer based architecture for HOI has emerged, which directly predicts the HOI triplets in an end-to-end fashion (Image → HOI). Motivated by various inference paths for HOI detection, we propose cross-path consistency learning (CPC), which is a novel end-to-end learning strategy to improve HOI detection for transformers by leveraging augmented decoding paths. CPC learning enforces all the possible predictions from permuted inference sequences to be consistent. This simple scheme makes the model learn consistent representations, thereby improving generalization without increasing model capacity. Our experiments demonstrate the effectiveness of our method, and we achieved significant improvement on V-COCO and HICO-DET compared to the baseline models. Our code is available at https://github.com/mlvlab/CPChoi.*

1. Introduction

Human-Object Interaction (HOI) detection is a holistic visual recognition task that includes detecting individual objects as <human, object>, while properly classifying the type of <interaction>. Previous HOI detectors [15, 31, 49, 52] were mainly built on object detection models. They commonly extend CNN-based object detectors [34, 42, 45] with an additional head for interaction classification, *e.g.*, humans and objects are detected first, and their interaction is associated subsequently.

To alleviate the high computation cost of such two-stage HOI detection methods, one-stage models [22, 33, 52] have been proposed for faster detection. These models perform interaction prediction and object detection in parallel. They compensate for their lower performance with auxiliary predictions for the HOI subsets, *i.e.*, auxiliary predictions for subset <human, interaction> or <object, interaction> may help HOI prediction through post-processing. However, these works demand different network architectures for each auxiliary prediction, due to strict disciplines for each network's input. Hence, to introduce flexibility, transformer-based architectures [6, 23, 46, 48] have recently been adopted for HOI detection. They reformulate the HOI detection problem as a direct set prediction building on DETR [4].

Motivated by various inference paths in HOI detectors, we propose a simple yet effective method to train HOI transformers. We augment the decoding paths with respect to the possible prediction sequences of HOI triplets. Then, with the cascade structure of transformers, an input query is sequentially decoded into auxiliary sub-task outputs and the final output. The stage of each augmented paths stage shares a decoder, in a multi-task learning fashion. We further improve our method to leverage the augmented decod-



(a) cycle consistency (b) cross-task consistency (c) cross-path consistency

Figure 1. **Comparison on the variants of consistencies**. The black line refers to the main task function $f_{\mathcal{X}\mathcal{Y}}$, and the red, blue lines refer to the pair of tasks trained to be consistent with each other. (a) Cycle consistency enforces the composite function of $f_{\mathcal{Y}\mathcal{X}} \circ f_{\mathcal{X}\mathcal{Y}}$ to be consistent with identity function $f_{\mathcal{X}\mathcal{X}}$. (b) Cross-task consistency requires an auxiliary pretrained network $f_{\mathcal{Y}\mathcal{Y}_a}$, represented in dashed lines, to give consistent outputs across tasks. (c) Cross-path consistency does not require task-specific pretrained networks. The output of main task function $f_{\mathcal{X}\mathcal{Y}}$ should be consistent with the composition of the outputs from sub-task functions $f_{\mathcal{X}\mathcal{Y}_s}$ and $f_{\mathcal{Y}_s\mathcal{Y}} \circ f_{\mathcal{X}\mathcal{Y}_s}$.

*corresponding author.

ing paths by enforcing the outputs from the various paths to be consistent. Accordingly, we propose **Cross-Path Consistency (CPC) Learning**, which aims to predict HOI triplets regardless of inference sequences.

Similar to cross-*task* consistency [55], cross-*path* consistency retains *inference path invariance*. However, cross-path consistency learning does not require additional pre-trained networks. In contrast to cross-task consistency, which demands an auxiliary network to train the main task $\mathcal{X} \rightarrow \mathcal{Y}$ (Figure 1-(b)), cross-path consistency defines an auxiliary domain $\mathcal{Y}_s$ in between $\mathcal{X}$ and $\mathcal{Y}$ (Figure 1-(c)). In other words, the main task $\mathcal{X} \rightarrow \mathcal{Y}$ (*i.e.*, Image→HOI) is divided into subtasks $\mathcal{X} \rightarrow \mathcal{Y}_s$ and $\mathcal{Y}_s \rightarrow \mathcal{Y}$ (*e.g.*, Image→HO→I). The main task function $f_{\mathcal{XY}}$ is then trained by enforcing its output and the composition of sub-task predictions to be consistent. Moreover, cross-path consistency learning is temporarily adopted for training only.

Our training strategy can be generalized to any transformer based architecture, and can be applied in an end-to-end method. Extensive experiments show that HOI transformers trained with CPC learning strategy achieves substantial improvements in two popular HOI detection benchmarks: V-COCO and HICO-DET. The contribution of this work can be summarized as the followings:

- We propose **Cross-Path Consistency** (CPC) learning, which is a novel end-to-end learning strategy to improve transformers for HOI detection leveraging various inference paths. In this learning scheme, we use **Decoding-Path Augmentation** to generate various inference paths which are compositions of subtasks with a shared decoder for effective training.
- Our training scheme achieves substantial improvements on V-COCO and HICO-DET without increasing *model capacity* and *inference time*.

2. Related Works

2.1. Human Object Interaction Detection

Human-Object Interaction (HOI) detection has been proposed in [16]. Later, human-object detectors have been improved using human or instance appearance and their spatial relationship [12, 15, 25]. On the other hand, graph-based approaches [11, 44, 49, 51] have been proposed to clarify the action between the `<human, object>` pair.

HOI detection models based on only visual cues often suffer from the lack of contextual information. Thus, recent works utilize external knowledge to improve the quality of HOI detection. Human pose information extracted from external models [3, 7, 19, 28] or linguistic priors and knowledge graph models show meaningful improvement in performance [14, 18, 31, 36, 37, 43, 54, 57, 58].

Since the majority of the previous works are based on two-stage methods with slower inference time, attempts for faster HOI detection by introducing simple end-to-end multi-layer perceptrons [17], or directly detecting interaction points [33, 52], or union regions [20, 22, 30] have been suggested.

2.2. Transformers in Computer Vision

Transformer has become the state-of-the-art method in many computer vision tasks. In image classification, [9] has shown competitive performance on ImageNet without any convolution layers. DeiT [48] applied knowledge distillation to data-efficiently train the vision transformer. To extract multi-scale image features, Swin Transformer [38] proposed shifted window based self-attention modules that effectively aggregate small patches to increase the receptive field. In the object detection task, DETR [4] has proposed an end-to-end framework eliminating the need for hand-designed components. DETR's bipartite matching loss between the predicted set and the ground truth labels enables direct set prediction at inference. Recently, DETR's late convergence problem has been tackled in [13, 40, 62].

Inspired by DETR, transformer-based HOI (Human-Object Interaction) detectors [6, 8, 23, 46, 63] have been recently proposed. HOI transformer models have two types of structure, one decoder model and the two decoder model. The one-decoder model which follows the structure of DETR [4] predicts triplets from the output of a single decoder. QPIC [46] and HoiT [63] are one-decoder models that output `<human, object, interaction>` triplets directly with multiple interaction detection heads. Two-decoder models use two transformer decoders to output distinctive targets. For instance, HOTR [23] and AS-NET [6] are composed of an instance decoder that outputs object and an interaction decoder that outputs interaction. In contrast to previous works that are trained with a single inference path, our model learns with the augmented decoding paths. Also, our framework can be applied to any transformer-based model. More explanation of HOI transformers are in Section 3.1.

2.3. Consistency Learning in Vision

Consistency constraints applied to many computer vision topics have been extensively studied. In semi-supervised learning, consistency regularization is widely used to train the model to be invariant to input noise. Label consistency methods [27, 41, 47, 53] augment or perturb an input image and apply consistency loss between model predictions. CDS [21] explored object detection in a semi-supervised setting with classification and localization consistency regularization. Also, consistency regularization in cyclic form is commonly used in generative models [61], image matching [59, 60], temporal correspondence [10], and in many

other domains.

Comparison with Consistency Learning Our consistency training scheme is relevant to cross-task consistency learning [55]. Cross-task consistency learning is based on inference-path invariance, where the predictions should be consistent regardless of the inference paths.

As shown in Figure 1 (b), cross-*task* consistency learning uses an auxiliary task $\mathcal{Y} \rightarrow \mathcal{Y}_a$ to train the main task function $f_{\mathcal{X}\mathcal{Y}}$, *i.e.*, given x from the query domain, and y from target domain $\mathcal{Y}$, predictions of $f_{\mathcal{Y}\mathcal{Y}_a} \circ f_{\mathcal{X}\mathcal{Y}}(x)$ and $f_{\mathcal{Y}\mathcal{Y}_a}(y)$ are expected to be consistent. Different from cross-*task* consistency, our cross-*path* consistency learning (Figure 1 (c)) trains the main task function $f_{\mathcal{X}\mathcal{Y}}$ by enforcing the prediction of target domain $\mathcal{Y}$ of $f_{\mathcal{X}\mathcal{Y}}$ and $f_{\mathcal{Y}_s\mathcal{Y}} \circ f_{\mathcal{X}\mathcal{Y}_s}$, where auxiliary domain $\mathcal{Y}_s$ is decomposed from the target domain $\mathcal{Y}$, to be consistent. Also, while cross-*task* consistency learning requires the mapping function $f_{\mathcal{Y}\mathcal{Y}_a}$ to be pretrained to avoid suboptimal training with the noisy estimator, cross-*path* consistency learning does not demand any task-specific pre-trained networks since the auxiliary domain $\mathcal{Y}_s$ is part of the target domain $\mathcal{Y}$. Details for our framework is described in section 3.2.

3. Method

In this section, we present our novel end-to-end training strategy for Transformers with **cross-path consistency** in Human-Object Interaction Detection. The training strategy includes 1) augmenting the decoding path and 2) consistency regularization between predictions of multiple decoding paths. Before discussing our training strategy, we briefly summarize transformers in Human-Object Interaction detection.

3.1. Transformer in HOI detection

HOI transformers are commonly extended upon DETR [4], which is composed of a CNN backbone followed by the encoder-decoder architecture of Transformer [1]. The CNN backbone first extracts a *locally* aggregated feature map $f \in \mathbb{R}^{H' \times W' \times D}$ from input image $x \in \mathbb{R}^{H \times W \times 3}$. Then, the feature map f is passed into the encoder to *globally* aggregate features via the self-attention mechanism, resulting in the encoded feature map $X \in \mathbb{R}^{H' \times W' \times D}$. At a decoding stage, a decoder takes learnable query embeddings $q \in \mathbb{R}^{N \times D}$ and outputs $e \in \mathbb{R}^{N \times D}$ by interacting with encoded feature map X through cross-attention. The outputs are converted to final HOI predictions (*i.e.*, human, object, interaction) by read-out functions, which are generally feed-forward networks.

Training Transformers for detection entails matching between predictions and ground truth labels since Transformers provide detections as set predictions. To compute losses, the Hungarian algorithm [26] is used to associate detections with ground truth labels. The predictions unmatched with ground truth labels are considered as no object or no interactions. In general, HOI transformers can be categorized into two groups based on human/object localization schemes. [46, 63] directly predict the box coordinates of human and object from an HOI prediction. But this causes problems that human or object can be redundantly predicted by multiple query embeddings and the localizations of the same object often differ across HOI triplet predictions. To address these problems, [6, 23] propose parallel architectures to perform interaction detection separately from object detection.

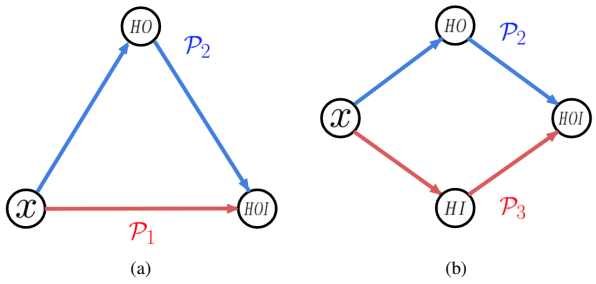


Figure 2. **Cross-path consistency for HOI detection**. (a) Main task path $\mathcal{P}_1$ should be consistent with each augmented path. *e.g.* path $\mathcal{P}_2$. (b) Augmented paths should be consistent with one another. *e.g.* path $\mathcal{P}_2$ and $\mathcal{P}_3$.

3.2. Decoding-Path Augmentation

We observe that HOI detection can be achieved by various sequences of predictions. For instance, CNN-based HOI detection models [5, 11, 15, 17] first detect instances (human and object) and then predict interactions between the instances, *i.e.*, $x \rightarrow$ HO $\rightarrow$ I, where x is an input image and H, O, I are predictions for human, object, interaction, respectively. On the other hand, the HOI Transformers by [6,23,46,63] directly predict HOI triplets, *i.e.*, $x \rightarrow$ HOI. Inspired by Cross-Task Consistency [56] and this observation, we propose **decoding-path augmentation** to generate various decoding paths (or prediction paths) and impose consistency regularization. Decoding-path augmentation for Transformers in HOI detection can be easily achieved by partially decoded HOI predictions. Furthermore, sharing decoders across paths is beneficial in terms of knowledge sharing.

In our experiments, we consider four decoding paths as follows:

$$\left.\begin{aligned} \mathcal{P}_1 &= x \rightarrow \text{HOI} \\ \mathcal{P}_2 &= x \rightarrow \text{HO} \rightarrow \text{I} \\ \mathcal{P}_3 &= x \rightarrow \text{HI} \rightarrow \text{O} \\ \mathcal{P}_4 &= x \rightarrow \text{OI} \rightarrow \text{H} \end{aligned}\right\} \textit{Augmented.} \tag{1}$$

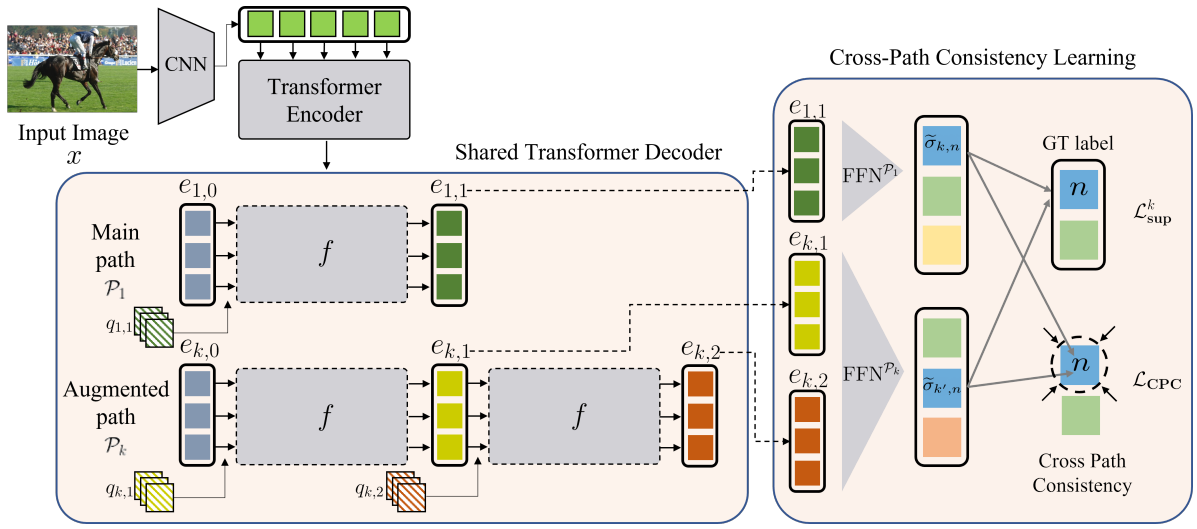


Figure 3. **The overall process of Cross-Path Consistency Learning**. The encoded image features are passed into the shared decoder with multiple inference paths $\{\mathcal{P}_1, ..., \mathcal{P}_{k-1}, \mathcal{P}_k\}$. Each path is augmented based on the decoding-path augmentation to generate various sequences of inference paths (see Section 3.2). To avoid clutter, we visualize only the main path $\mathcal{P}_1$ and an augmented path $\mathcal{P}_k$. The main path $\mathcal{P}_1$ consists of a single decoding stage, and the augmented path $\mathcal{P}_k$ is a composition of decoding stages; all f blocks share parameters. Given queries q a learnable position embeddings, each decoder extracts output embeddings denoted as $e_{1,1}$, $e_{k,1}$, and $e_{k,2}$. Then, each of the output embeddings is fed into the readout function FFN to predict each HOI element *i.e.* `<human, object, interaction>`. With Cross-Path Consistency Learning (Section 3.3), all the outputs supervised with the same ground truth label are trained to be consistent regardless of their inference paths. Cross-Matching is used to match the queries that are considered to be consistent by leveraging ground truth label. Along with the supervision loss $\mathcal{L}^k_{\text{sup}}$ for all paths $\mathcal{P}_k$, cross-path consistency loss $\mathcal{L}_{\text{CPC}}$ is added to our final loss.

Each decoding stage of path $\mathcal{P}_k$ can be written as:

$$\begin{aligned} e_{k,1} &= f(e_{k,0} + q_{k,1},\ X), \\ e_{k,2} &= f(e_{k,1} + q_{k,2},\ X), \end{aligned} \tag{2}$$

where $q_{k,j}$, $e_{k,j}$ denote learnable query and output embeddings on k^{th} path at j^{th} decoding stage. The decoder f is shared across all paths and stages. The $e_{k,0}$ above is dummy output embeddings set to zeros since there is no 0-th stage, see Figure 3. Each decoding stage and path use a separate readout function FFN to translate the output embeddings into HOI instance predictions. For example, on $\mathcal{P}_2 : x \rightarrow \text{HO} \rightarrow \text{I}$, at stage 1 $e_{2,1}$ is read out by $\text{FFN}_h^{\mathcal{P}_2}$ and $\text{FFN}_o^{\mathcal{P}_2}$ to predict bounding boxes of human and object respectively. Prediction for HOI element $m \in \{h, o, act\}$ in each k^{th} path at j^{th} decoding stage can be written as $\hat{y}_k^m = \text{FFN}_m^{\mathcal{P}_k}(e_{k,j})$.

3.3. Cross-Path Consistency Learning

We now present our Cross-Path Consistency Learning framework (CPC) that imposes consistency regularization between predictions from different decoding paths as shown in Figure 2. Learning with CPC leads better generalization without any additional data or labels.

Cross-Path Consistency. We explain our consistency learning scheme with an exemplary case of main path $\mathcal{P}_1$ and augmented path $\mathcal{P}_2$ given as

$$\begin{aligned} \mathcal{P}_1 &: x \rightarrow \text{HOI} \\ \mathcal{P}_2 &: x \rightarrow \text{HO} \rightarrow \text{I}. \end{aligned} \tag{3}$$

Here, the main path $\mathcal{P}_1$ is the HOI transformers' original inference path. In path $\mathcal{P}_2$, human and object detection logits $\hat{y}_2^h$ and $\hat{y}_2^o$ are obtained reading out $e_{2,1}$, which is the output embeddings on path 2 at stage 1. Then, the interaction logit $\hat{y}_2^{act}$ is obtained after another subsequent decoder pass defined as $f_{2,2}$. The corresponding inference scheme of $\mathcal{P}_2$ can be written in more formal terms:

$$\begin{aligned} \hat{y}_2^h &= \text{FFN}_h^{\mathcal{P}_2}(f_{2,1}(X)) \\ \hat{y}_2^o &= \text{FFN}_o^{\mathcal{P}_2}(f_{2,1}(X)) \\ \hat{y}_2^{act} &= \text{FFN}_{act}^{\mathcal{P}_2}(f_{2,2} \circ f_{2,1}(X)) \end{aligned} \tag{4}$$

In (4), input arrays for f other than feature map X were omitted for simplicity.

With the predictions, we impose regularization to make the outputs from path $\mathcal{P}_1$ and path $\mathcal{P}_2$ consistent. Note that HOI detections from $\mathcal{P}_2$ consist of both final and intermediate decoder outputs. To this end, we define the loss function $\mathcal{L}_{\mathcal{P}_1\mathcal{P}_2}$ by aggregating losses from multiple augmented paths to enforce consistency. The loss function is given as:

$$\begin{aligned} \mathcal{L}_{\mathcal{P}_1\mathcal{P}_2} = \lambda_h \cdot \mathcal{L}_h(\hat{y}_1^h, \hat{y}_2^h) + \lambda_o \cdot \mathcal{L}_o(\hat{y}_1^o, \hat{y}_2^o) \\ + \lambda_{act} \cdot \mathcal{L}_{act}(\hat{y}_1^{act}, \hat{y}_2^{act}), \end{aligned} \tag{5}$$

where $\hat{y}_1^h$, $\hat{y}_1^o$ and $\hat{y}_1^{act}$ are the output from the main path $\mathcal{P}_1$ and λ are the loss weights. In our experiments, softmax-type outputs use Jensen-Shannon divergence (JSD) for consistency loss to give loss to each path symmetrically, while outputs followed by sigmoid, *e.g.*, box regression, multi-label action classes, take the Mean-Squared Error loss. More details on type-specific loss functions are in the supplement.

In the case of other path pairs, loss is computed in the same manner. The final loss should thus incorporate all possible pairs. Then, the cross-path consistency (CPC) loss can be written as:

$$\mathcal{L}_{\textbf{CPC}} = \frac{1}{S} \sum_{(k,k') \in \mathcal{K}} \mathcal{L}_{\mathcal{P}_k \mathcal{P}_{k'}} \tag{6}$$

where $\mathcal{K}$ denotes the set of all possible path pairs, and S refers to the size of set $\mathcal{K}$, *i.e.* the number of path combinations.

Cross Matching. Cross-path consistency learning encourages outputs from different paths to be consistent. However, since the outputs from a path are given as a set, we first need to resolve correspondence to specify the pairs of predictions to enforce consistency. We present **cross matching**, a simple method that tags each instance with its corresponding ground truth label. The instances tagged with the same label are paired to compute consistency loss. On the other hand, if an instance is not matched with any of the paths' output, we simply exclude the instance from consistency learning treating it as *no object* or *no interaction*. Our cross-path consistency loss is introduced below.

Let $\sigma_k(i)$ denote the index of the ground truth label that matches the i^{th} query in the k^{th} path. We define $\sigma_k^{-1}(n)$ as the query index of path $\mathcal{P}_k$ which is matched with the ground truth index n. To avoid clutter, we use $\tilde{\sigma}_{k,n}$ as a shorthand notation for $\sigma_k^{-1}(n)$. The outputs from different paths with the same ground-truth label should be consistent. For example, $\hat{y}^m_{k,\tilde{\sigma}_{k,n}}$ and $\hat{y}^m_{k,\tilde{\sigma}_{k',n}}$ which are predictions for m from $\mathcal{P}_k$ and $\mathcal{P}_{k'}$ with the same ground-truth index n should be consistent.

Cross-path consistency loss between output predictions from $\mathcal{P}_k$ and $\mathcal{P}_{k'}$ with the same ground-truth with index n is defined as follows:

$$\begin{aligned} \mathcal{L}^n_{\mathcal{P}_k \mathcal{P}_{k'}} = \; & \lambda_h \cdot \mathcal{L}_h\big(\hat{y}^h_{k,\tilde{\sigma}_{k,n}}, \hat{y}^h_{k',\tilde{\sigma}_{k',n}}\big) \\ & + \lambda_o \cdot \mathcal{L}_o\big(\hat{y}^o_{k,\tilde{\sigma}_{k,n}}, \hat{y}^o_{k',\tilde{\sigma}_{k',n}}\big) \\ & + \lambda_{act} \cdot \mathcal{L}_{act}\big(\hat{y}^{act}_{k,\tilde{\sigma}_{k,n}}, \hat{y}^{act}_{k',\tilde{\sigma}_{k',n}}\big) \end{aligned} \tag{7}$$

Final Loss The final cross-path consistency loss for all $\mathcal{P}_k$ is derived as,

$$\mathcal{L}_{\textbf{CPC}} = \frac{1}{S \cdot \mathcal{N}} \sum_{n=1}^{\mathcal{N}} \sum_{(k,k') \in \mathcal{K}} \mathcal{L}^n_{\mathcal{P}_k \mathcal{P}_{k'}} \tag{8}$$

where $\mathcal{N}$ is the number of ground truth labels. Then, the final form of our training loss $\mathcal{L}$ is defined by

$$\mathcal{L} = \sum_k \mathcal{L}^k_{\textbf{sup}} + w(t) \cdot \mathcal{L}_{\textbf{CPC}}, \tag{9}$$

where $\mathcal{L}^k_{\textbf{sup}}$ is the supervision loss for each path $\mathcal{P}_k$ and $w(t)$ is a ramp-up function [2, 27, 47] for stable training. Our overall framework is illustrated in Figure 3.

4. Experiments

In this section, we empirically evaluate the effectiveness of our cross-path consistency learning with HOI transformers. Our experiments are conducted on public HOI detection benchmark datasets: **V-COCO** and **HICO-DET**. We first briefly introduce the datasets and provide implementation details. Our extensive experiments demonstrate that our training strategy renders significant improvement on the baseline models without additional parameters or inference time.

4.1. Dataset

V-COCO [16] is a subset of the COCO dataset [35] which contains 5,400 `trainval` images and 4,946 `test` images. V-COCO is annotated with 29 common action classes. For evaluation of the V-COCO dataset, we report the mAP metric over 25 interactions for two scenarios, The first scenario includes a prediction of occluded objects and is evaluated with respect to AP_{role1}. On the other hand, the second scenario does not contain such cases, and performance is measured in AP_{role2}.

HICO-DET [5] is a subset of the HICO dataset which has more than 150K annotated instances of human-object pairs in 47,051 images (37,536 for training and 9,515 for testing). It is annotated with 600 `<interaction, object>` instances. There are 80 unique object types, identical to the COCO object categories, and 117 unique interaction verbs. For evaluation of the HICO-DET, we report the mAP over three different set categories: (1) all 600 HOI categories in HICO (Full), (2) 138 HOI categories with less than 10 training samples (Rare), and (3) 462 HOI categories with more than 10 training samples (Non-Rare).

4.2. Implementation Details

Training In our experiment, QPIC [46] and HOTR [23] were used as the baseline for the HOI transformer respectively. During training, we initialize the network with pre-trained DETR [4] on MS-COCO with a Resnet-50 backbone. For all decoding paths, parameters of the model are shared except for stage-wise queries and feedforward networks.

All our experiments using consistency regularization are trained for 90 epochs and the learning rate is decayed at the 60-th epoch by a factor of 0.1. As an exception, HOTR is trained up to 50 epochs and the learning rate is decayed at epoch 30 by a factor of 0.1 for HICO-DET. Following the original training schemes in QPIC and HOTR, we freeze the encoder and backbone for HOTR, whereas unfreeze those for QPIC. We use the AdamW [39] optimizer with a batch size of 16, and the initial learning rates for the transformer and backbone parameters are set to 10^{-4} and 10^{-5} respectively, and weight decay is set to 10^{-4}. All experiments are trained on 8 V100 GPUs.

We re-implement the result of QPIC and HOTR on V-COCO [16] since our reproduction results are quite different from the official ones in the paper. For a fair comparison, all the loss coefficients overlapping between baselines and our training strategy are identical to the ones reported in the paper [23, 46]. Details for hyperparameters relevant to our training strategy are reported in the supplementary material.

Inference We mainly use $\mathcal{P}_1$ ($x \rightarrow$ HOI) for inference to compare with the baseline models without increasing the number of parameters. Also, we report the results of other inference paths in our ablation studies.

4.3. Comparison with HOI transformer

We evaluate the effectiveness of our method compared to the existing HOI transformers. All experiments are reported with the main path $\mathcal{P}_1$ that infers HOI triplets by a single decoding stage ($x \rightarrow$ HOI) which is identical to the original HOI transformer. As shown in Table 1, our CPC training strategy significantly outperforms on two baselines, HOTR [23] and QPIC [46]. In the V-COCO dataset, the experiment shows improvement in performance by a considerable margin of 0.9 mAP for QPIC in AP_{role1}, and 1.8 mAP for HOTR. For AP_{role2}, QPIC and HOTR gain improvement by 0.9 mAP and 1.9 mAP respectively, similar to that of AP_{role1}.

In the HICO-DET dataset, our CPC learning with HOTR and QPIC outperforms all the evaluation categories of HICO-DET, except negligible degradation in the Non-Rare category on HOTR. Results on rare class on the HICO-DET are improved by a significant margin of 1.29 mAP and 5.5 mAP for QPIC and HOTR respectively. In both models, we observe a more prominent performance improvement in the Rare category. This supports that our training strategy performs well on rarely seen examples. Our strategy improves the conventional HOI transformer models.

	V-COCO		HICO-DET		
Method	AP_{role1}	AP_{role2}	Full	Rare	Non-Rare
QPIC	62.2*	64.5*	29.07	21.85	31.23
QPIC + ours	**63.1**	**65.4**	**29.63**	**23.14**	**31.57**
HOTR	59.8*	64.9*	25.10	17.34	**27.42**
HOTR + ours	**61.6**	**66.8**	**26.16**	**22.84**	27.15

Table 1. **Comparison of our training strategy with vanilla HOI transformers on V-COCO and HICO-DET**. * signifies our results reproduced with the official implementation codes of QPIC and HOTR.

Method	Backbone	AP_{role1}	AP_{role2}
CNN-based HOI Detection Model			
InteractNet [15]	R50-FPN	40.0	48.0
iCAN [12]	R50	45.3	52.4
TIN [32]	R50	47.8	-
RPNN [58]	R50	-	47.5
Verb Embd. [54]	R50	45.9	-
PMFNet [50]	R50-FPN	52.0	-
PastaNet [31]	R50-FPN	51.0	57.5
VCL [20]	R50 L	48.3	-
UniDet [22]	R50-FPN	47.5	56.2
DRG [11]	R50-FPN	51.4	-
FCMNet [36]	R50	53.1	-
ConsNet [37]	R50-FPN	53.2	-
PDNet [57]	R50-FPN	53.3	-
IDN [30]	R50	53.3	60.3
GPNN [44]	R152	44.0	-
IPNet [52]	H.G.104	51.0	-
VSGNet [49]	R152	51.8	57.0
PDNet [57]	Res152	52.2	-
ACP [24]	Res152	53.0	-
Transformer-based HOI Detection Model			
HoiT [63]	R101	52.9	-
AS-Net [6]	R50	53.9	-
HOTR [23]	R50	55.2	64.4
HOTR+ **Ours**	R50	**61.6**	**66.8**
QPIC [46]	R50	58.8	61.0
QPIC+ **Ours**	R50	**63.1**	**65.4**

Table 2. **Comparison of performances on the V-COCO test set**. AP_{role1} and AP_{role2} denotes performances under Scenario 1 and Scenario 2 in V-COCO respectively.

4.4. Comparison with State-of-the-Art Methods

In Table 2 and Table 3, we compare previous HOI detection methods with ours. As demonstrated in the tables, our training strategy achieves the best performance among its peers. Table 2 shows the result on V-COCO dataset in both

Method	Detector	Backbone	Extra	Default		
				Full	Rare	Non Rare
CNN-based HOI Detection Model						
InteractNet [15]	COCO	R50-FPN	✗	9.94	7.16	10.77
iCAN [12]	COCO	R50	S	14.84	10.45	16.15
TIN [32]	COCO	R50	S+P	17.03	13.42	18.11
RPNN [58]	COCO	R50	P	17.35	12.78	18.71
PMFNet [50]	COCO	R50-FPN	S+P	17.46	15.65	18.00
No-Frills HOI [17]	COCO	R152	S+P	17.18	12.17	18.68
UnionDet [22]	COCO	R50-FPN	✗	14.25	10.23	15.46
DRG [11]	COCO	R50-FPN	S+L	19.26	17.74	19.71
VCL [20]	COCO	R50	S	19.43	16.55	20.29
FCMNet [36]	COCO	R50	S+P	20.41	17.34	21.56
ACP [24]	COCO	R152	S+P	20.59	15.92	21.98
DJ-RN [29]	COCO	R50	S+V	21.34	18.53	22.18
ConsNet [37]	COCO	R50-FPN	S+L	22.15	17.12	23.65
PastaNet [31]	COCO	R50	S+P+L	22.65	21.17	23.09
IDN [30]	COCO	R50	S	23.36	22.47	23.63
GPNN [44]	COCO	R152	✗	13.11	9.41	14.23
IPNet [52]	COCO	HourGlass104	✗	19.56	12.79	21.58
VSGNet [49]	COCO	R152	S	19.80	16.05	20.91
PD-Net [57]	COCO	R152	S+P+L	20.81	15.90	22.28
Transformer-based HOI Detection Model						
HoiT [63]	HICO-DET	R50	✗	23.46	16.91	25.41
AS-Net [6]	HICO-DET	R50	✗	28.87	24.25	30.25
HOTR [23]	HICO-DET	R50	✗	25.10	17.34	**27.42**
HOTR+ **Ours**	HICO-DET	R50	✗	**26.16**	**22.84**	27.15
QPIC [46]	HICO-DET	R50	✗	29.07	21.85	31.23
QPIC+ **Ours**	HICO-DET	R50	✗	**29.63**	**23.14**	**31.57**

Table 3. **Performance comparison in HICO-DET**. For the Detector, COCO means that the detector is trained on COCO, while HICO-DET means that the detector is first trained on COCO and then fine-tuned on HICO-DET. The each letter in Extra column stands for S: Interaction Patterns (Spatial Correlations), P: Pose, L: Linguistic Priors, V: Volume.

Method	*Share Dec.*	*CPC*	$\mathcal{P}_1$	$\mathcal{P}_2$	$\mathcal{P}_3$	$\mathcal{P}_4$	***Average***
QPIC	✓	✓	**63.1**	**63.3**	**63.1**	**63.0**	**63.13** $\pm$ 0.05†
		✓	62.4	62.9	60.8	59.4	61.38 $\pm$ 1.38
	✓		60.7	60.7	59.9	58.1	59.85 $\pm$ 1.06
HOTR	✓	✓	**61.6**	61.5	**61.6**	**61.6**	**61.58** $\pm$ 0.02†
		✓	61.2	**61.6**	61.1	60.6	61.13 $\pm$ 0.36
	✓		60.6	60.6	61.2	60.6	60.75 $\pm$ 0.13

Table 4. **Ablation Study on our learning strategies**. Ablation results on shared decoder (*Share Dec.*), and Cross-Path Consistency (CPC) are demonstrated. For main path $\mathcal{P}_1$, and each augmented path $\mathcal{P}_2$, $\mathcal{P}_3$, $\mathcal{P}_4$, their performances are reported measured in mAP. They are evaluated on the V-COCO test set with respect to Scenario 1. The best performances for each path are highlighted in bold, and † refers to the case where the least standard deviation is observed.

AP_{role1} and AP_{role2}. In the V-COCO dataset, our method achieves outstanding performance of 63.1 mAP in AP_{role1} and 66.8 mAP in AP_{role2}. Also, the results on the HICO-DET dataset in Table 3 show that our CPC further improves the state-of-the-art models (*e.g.*, HOTR, and QPIC) in the default setting achieving 26.16 mAP and 29.63 mAP, respectively.

4.5. Ablation Study

We further discuss the effectiveness of our framework through a series of ablation studies. We first provide a path-wise analysis for our cross-path consistency learning method. The effect of our training technique components was tested on each path to validate our method. Subsequently, we analyze the impact of the number of augmented paths on the main task performance. We experimentally

prove the validity of our method by demonstrating the correlation between the number of paths and performance.

Efficiency of CPC. Table 4 presents ablation experiment results for all inference paths, $\mathcal{P}_1$, $\mathcal{P}_2$, $\mathcal{P}_3$, and $\mathcal{P}_4$. Path $\mathcal{P}_1$ is the main path, which we aim to boost performance with the rest of the augmented paths. We try ablating decoder sharing or cross-path consistency regularization one at a time to confirm each component's contribution to our training strategy. Note that all of our experiments are conducted with the encoder block shared across paths.

When our CPC training strategies are applied, QPIC and HOTR achieve an mAP of 63.1, and 61.6 on main path $\mathcal{P}_1$. When the decoder parameters are not shared, performance degradation in path $\mathcal{P}_1$ was observed for both baselines; a 0.7 mAP drop for QPIC, and 0.4 mAP drop for HOTR. On the other hand, when CPC regularization is left out while decoder parameters are shared, performance of QPIC and HOTR decreased by a large margin of 2.4 mAP and 1.0 mAP each. In terms of overall performance across all paths, the average mAP showed a similar trend for each experiment condition. The overall results support that our learning strategy improves generalization of base architectures, and boosts performance by sharing knowledge throughout paths and stages.

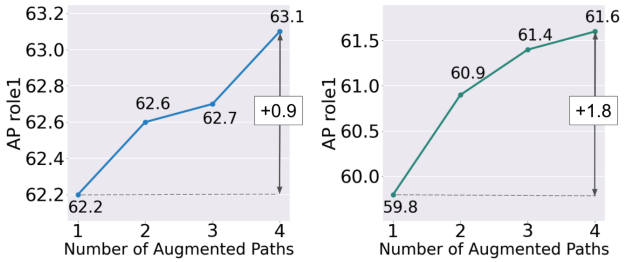
Interestingly, the standard deviation of all performances dramatically increases without both components. With unshared decoders, deviation increases by 1.33 for QPIC and 0.35 for HOTR. Also, when CPC regularization is removed, deviation increases by 1.01 for QPIC and 0.11 for HOTR. This implies that our training strategy with shared decoder and CPC leads to more stable training as well as consistent representations.

Impact of Augmented Paths. We explore how the number of augmented paths affects the performance of the main path $\mathcal{P}_1$ in V-COCO benchmark. Starting from $\mathcal{P}_1$, the augmented paths are gradually added with respect to mAP of Scenario 1 from Table 5, where each path is independently trained with default settings with no training techniques applied. We leverage the augmented path with better performance first, as performance of each model will serve as a lower bound for the ensemble of paths. Specifically, as shown in Table 5, both HOTR and QPIC showed better performance in the order of $\mathcal{P}_1$, $\mathcal{P}_2$, $\mathcal{P}_3$, and $\mathcal{P}_4$, when trained independently.

We compare the four cases where the augmented paths are gradually added in the corresponding order; *i.e.*, $\mathcal{P}_1$, $\mathcal{P}_1 + \mathcal{P}_2$, $\mathcal{P}_1 + \mathcal{P}_2 + \mathcal{P}_3$, and $\mathcal{P}_1 + \mathcal{P}_2 + \mathcal{P}_3 + \mathcal{P}_4$. As shown in Figure 4, performance is gradually improved as augmented paths are added. The ablation study evidences that regardless of each path performance, taking advantage of more paths bolsters the learning capability of our main task, and its performance builds up as the number of augmented paths increases.

Method	$\mathcal{P}_1$	$\mathcal{P}_2$	$\mathcal{P}_3$	$\mathcal{P}_4$	***Average***
QPIC	62.2	61.9	61.7	60.4	61.55 ± 0.69
HOTR	59.8	59.5	59.0	58.9	59.3 ± 0.37

Table 5. **Path-wise results on V-COCO.**



(a) QPIC (b) HOTR

Figure 4. **Ablation on the number of augmented paths**. As the number of augmented paths increases, main task performance increases accordingly.

5. Conclusion

We propose end-to-end Cross-Path Consistency learning for Human-Object Interaction detection. Through decoding-path augmentation, various decoder paths are generated which predict HOI triplets in permuted sequences. Then, consistency regularization is applied across paths to enforce the predictions to be consistent. Parameter sharing and cross-matching were introduced as well to enhance learning.

Our method is conceptually simple, and can be applied to a wide range of transformer architectures. Also, it does not require additional model capacity nor inference time. The substantial improvements on V-COCO and HICO-DET support our method's efficacy in various HOI detection tasks. Through further empirical studies, its capabilities to improve generalization and to encourage consistent representations are approved.

Acknowledgements This work was partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.2021-0-02312, Efficient Meta-learning Based Training Method and Multipurpose Multi-modal Artificial Neural Networks for Drone AI), (IITP-2022-2020-0-01819, the ICT Creative Consilience program); ETRI grant (22ZS1200, Fundamental Technology Research for Human-Centric Autonomous Intelligent System); and KakaoBrain corporation.

References

[1] Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser L., and Polosukhin I. Attention is all you need. In *NeurIPS*, 2017. 3

[2] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *NeurIPS*, 2019. 5

[3] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *CVPR*, 2017. 2

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 1, 2, 3, 6

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

[6] Mingfei Chen, Yue Liao, Si Liu, Zhiyuan Chen, Fei Wang, and Chen Qian. Reformulating hoi detection as adaptive set prediction. In *CVPR*, 2021. 1, 2, 3, 6, 7

[7] Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. Cascaded pyramid network for multi-person pose estimation. In *CVPR*, 2018. 2

[8] Qi Dong, Zhuowen Tu, Haofu Liao, Yuting Zhang, Vijay Mahadevan, and Stefano Soatto. Visual relationship detection using part-and-sum transformers with composite queries. In *ICCV*, 2021. 2

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2020. 2

[10] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. Temporal cycle-consistency learning. In *CVPR*, 2019. 2

[11] Chen Gao, Jiarui Xu, Yuliang Zou, and Jia-Bin Huang. DRG: dual relation graph for human-object interaction detection. In *ECCV*, 2020. 2, 3, 6, 7

[12] Chen Gao, Yuliang Zou, and Jia-Bin Huang. ican: Instance-centric attention network for human-object interaction detection. In *WACV*, 2018. 2, 6, 7

[13] Peng Gao, Minghang Zheng, Xiaogang Wang, Jifeng Dai, and Hongsheng Li. Fast convergence of detr with spatially modulated co-attention. In *ICCV*, 2021. 2

[14] Nikolaos Gkanatsios, Vassilis Pitsikalis, Petros Koutras, Athanasia Zlatintsi, and Petros Maragos. Deeply supervised multimodal attentional translation embeddings for visual relationship detection. In *ICCV*, 2019. 2

[15] Georgia Gkioxari, Ross Girshick, Piotr Dollár, and Kaiming He. Detecting and recognizing human-object interactions. In *CVPR*, 2018. 1, 2, 3, 6, 7

[16] Jitendra Gupta, Saurabh Malik. Visual semantic role labeling. In *CVPR*, 2015. 2, 5, 6

[17] Tanmay Gupta, Alexander Schwing, and Derek Hoiem. No-frills human-object interaction detection: Factorization, layout encodings, and training techniques. In *ICCV*, 2019. 2, 3, 7

[18] Tanmay Gupta, Kevin Shih, Saurabh Singh, and Derek Hoiem. Aligned image-word representations improve inductive transfer across vision-language tasks. In *ICCV*, 2017. 2

[19] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 2

[20] Zhi Hou, Xiaojiang Peng, Yu Qiao, and Dacheng Tao. Visual compositional learning for human-object interaction detection. In *ECCV*, 2020. 2, 6, 7

[21] Jisoo Jeong, Seungeui Lee, Jeesoo Kim, and Nojun Kwak. Consistency-based semi-supervised learning for object detection. In *NeurIPS*, 2019. 2

[22] Bumsoo Kim, Taeho Choi, Jaewoo Kang, and Hyunwoo J. Kim. Uniondet: Union-level detector towards real-time human-object interaction detection. In *ECCV*, 2020. 1, 2, 6, 7

[23] Bumsoo Kim et al. Hotr: End-to-end human-object interaction detection with transformers. In *CVPR*, 2021. 1, 2, 3, 6, 7

[24] Dong-Jin Kim, Xiao Sun, Jinsoo Choi, Stephen Lin, and In So Kweon. Detecting human-object interactions with action co-occurrence priors. In *ECCV*, 2020. 6, 7

[25] Alexander Kolesnikov, Alina Kuznetsova, Christoph H Lampert, and Vittorio Ferrari. Detecting visual relationships using box attention. In *ICCV*, 2019. 2

[26] Harold W Kuhn. The hungarian method for the assignment problem. *NRL*, 1955. 3

[27] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *ICLR*, 2017. 2, 5

[28] Jiefeng Li, Can Wang, Hao Zhu, Yihuan Mao, Hao-Shu Fang, and Cewu Lu. Crowdpose: Efficient crowded scenes pose estimation and a new benchmark. In *CVPR*, 2019. 2

[29] Yong-Lu Li, Xinpeng Liu, Han Lu, Shiyi Wang, Junqi Liu, Jiefeng Li, and Cewu Lu. Detailed 2d-3d joint representation for human-object interaction. In *CVPR*, 2020. 7

[30] Yong-Lu Li, Xinpeng Liu, Xiaoqian Wu, Yizhuo Li, and Cewu Lu†. Hoi analysis: Integrating and decomposing human-object interaction. In *NeurIPS*, 2020. 2, 6, 7

[31] 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 *CVPR*, 2020. 1, 2, 6, 7

[32] Yong-Lu Li, Siyuan Zhou, Xijie Huang, Liang Xu, Ze Ma, Hao-Shu Fang, Yanfeng Wang, and Cewu Lu. Transferable interactiveness knowledge for human-object interaction detection. In *CVPR*, 2019. 6, 7

[33] Yue Liao, Si Liu, Fei Wang, Yanjie Chen, Chen Qian, and Jiashi Feng. Ppdm: Parallel point detection and matching for real-time human-object interaction detection. In *CVPR*, 2020. 1, 2

[34] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 1

[35] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence

Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 5

[36] Yang Liu, Qingchao Chen, and Andrew Zisserman. Amplifying key cues for human-object-interaction detection. In *ECCV*, 2020. 2, 6, 7

[37] Ye Liu, Junsong Yuan, and Chang Wen Chen. Consnet: Learning consistency graph for zero-shot human-object interaction detection. In *ACMMM*, 2020. 2, 6, 7

[38] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 2

[39] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 6

[40] Depu Meng, Xiaokang Chen, Zejia Fan, Gang Zeng, Houqiang Li, Yuhui Yuan, Lei Sun, and Jingdong Wang. Conditional detr for fast training convergence. In *ICCV*, 2021. 2

[41] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *TPAMI*, 2018. 2

[42] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, 2016. 1

[43] Julia Peyre, Ivan Laptev, Cordelia Schmid, and Josef Sivic. Detecting unseen visual relations using analogies. In *ICCV*, 2019. 2

[44] Siyuan Qi, Wenguan Wang, Baoxiong Jia, Jianbing Shen, and Song-Chun Zhu. Learning human-object interactions by graph parsing neural networks. In *ECCV*, 2018. 2, 6, 7

[45] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: towards real-time object detection with region proposal networks. *TPAMI*, 2016. 1

[46] Masato Tamura et al. Qpic: Query-based pairwise human-object interaction detection with image-wide contextual information. In *CVPR*, 2021. 1, 2, 3, 6, 7

[47] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*, 2017. 2, 5

[48] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021. 1, 2

[49] Oytun Ulutan, ASM Iftekhar, and BS Manjunath. Vsgnet: Spatial attention network for detecting human object interactions using graph convolutions. In *CVPR*, 2020. 1, 2, 6, 7

[50] Bo Wan, Desen Zhou, Yongfei Liu, Rongjie Li, and Xuming He. Pose-aware multi-level feature network for human object interaction detection. In *ICCV*, 2019. 6, 7

[51] Hai Wang, Wei shi Zheng, and Ling Yingbiao. Contextual heterogeneous graph network for human-object interaction detection. In *ECCV*, 2020. 2

[52] Tiancai Wang, Tong Yang, Martin Danelljan, Fahad Shahbaz Khan, Xiangyu Zhang, and Jian Sun. Learning human-object interaction detection using interaction points. In *CVPR*, 2020. 1, 2, 6, 7

[53] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. In *NeurIPS*, 2020. 2

[54] Bingjie Xu, Yongkang Wong, Junnan Li, Qi Zhao, and Mohan S Kankanhalli. Learning to detect human-object interactions with knowledge. In *CVPR*, 2019. 2, 6

[55] Amir R Zamir et al. Robust learning through cross-task consistency. In *CVPR*, 2020. 2, 3

[56] Ying Zhang, Tao Xiang, Timothy M Hospedales, and Huchuan Lu. Deep mutual learning. In *CVPR*, 2018. 3

[57] Xubin Zhong, Changxing Ding, Xian Qu, and Dacheng Tao. Polysemy deciphering network for robust human-object interaction detection. In *ECCV*, 2020. 2, 6, 7

[58] Penghao Zhou and Mingmin Chi. Relation parsing neural network for human-object interaction detection. In *ICCV*, 2019. 2, 6, 7

[59] Tinghui Zhou, Philipp Krahenbuhl, Mathieu Aubry, Qixing Huang, and Alexei A Efros. Learning dense correspondence via 3d-guided cycle consistency. In *CVPR*, 2016. 2

[60] Tinghui Zhou, Yong Jae Lee, Stella X. Yu, and Alexei A. Efros. Flowweb: Joint image set alignment by weaving consistent, pixel-wise correspondences. In *CVPR*, 2015. 2

[61] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017. 2

[62] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *ICLR*, 2020. 2

[63] 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 *CVPR*, 2021. 2, 3, 6, 7