Supplementary Material for QPIC: Query-Based Pairwise Human-Object Interaction Detection with Image-Wide Contextual Information

Masato Tamura[1], Hiroki Ohashi[2], and Tomoaki Yoshinaga[1]

[1]Lumada Data Science Lab., Hitachi, Ltd.
[2]Center for Technology Innovation - Artificial Intelligence, Hitachi, Ltd.
{*masato.tamura.sf, hiroki.ohashi.uo, tomoaki.yoshinaga.xc*}*@hitachi.com*

A. Supplementary V-COCO Settings

As mentioned in the main manuscript, the images of the V-COCO dataset are split into three sets: a training set, validation set, and testing set. Following previous works, the training and validation sets are combined to train QPIC.

For calculating the mAP, 5 action classes out of the 29 classes are excluded from the evaluation following [4]. This is because four of the excluded action classes ("run", "smile", "stand", and "walk") are the action without an object, and one of them ("point") has an insufficient number of samples.

B. Supplementary Implementation Note

As usual training, we use data augmentation to alleviate over-fitting. We use random horizontal flipping augmentation, scale augmentation, random crop augmentation, which are used in DETR's training [1], and color augmentation, which is used in PPDM's training [10].

Since each layer of a transformer decoder output its own set of embeddings $\boldsymbol{D} = \{\boldsymbol{d}_i | \boldsymbol{d}_i \in \mathbb{R}^{D_c}\}_{i=1}^{N_q}$, the loss calculation described in Sec. 3.2 of the main manuscript can be conducted for each layer. Following the DETR's training [1], these auxiliary losses are calculated to optimize QPIC. To calculate the losses, FFNs are added on top of each decoder layer's output. Note that the parameters of the FNNs are shared among all the decoder layers,

In the evaluation time, the second highest scoring class and confidence of the object-class prediction $\hat{\boldsymbol{c}}_i$ are used to generate the detection result if $\hat{\boldsymbol{c}}_i$ has the highest score in "no pair" class. This is the technique used in [1] to optimize the mAPs.

C. Additional List of Comparison

Table 1 and Table 2 show the additional list of the comparison against state-of-the-art on HICO-DET [2] and V-COCO [5], respectively. Six methods (PMFNet [15], Wang *et al*. [16], In-GraphNet [18], ACP [8], PD-Net [19], and DJ-RM [9]) are additionally compared in these tables. As stated in Sec. 4.3 of the main manuscript, our QPIC significantly outperforms conventional two- and single-stage methods on both datasets.

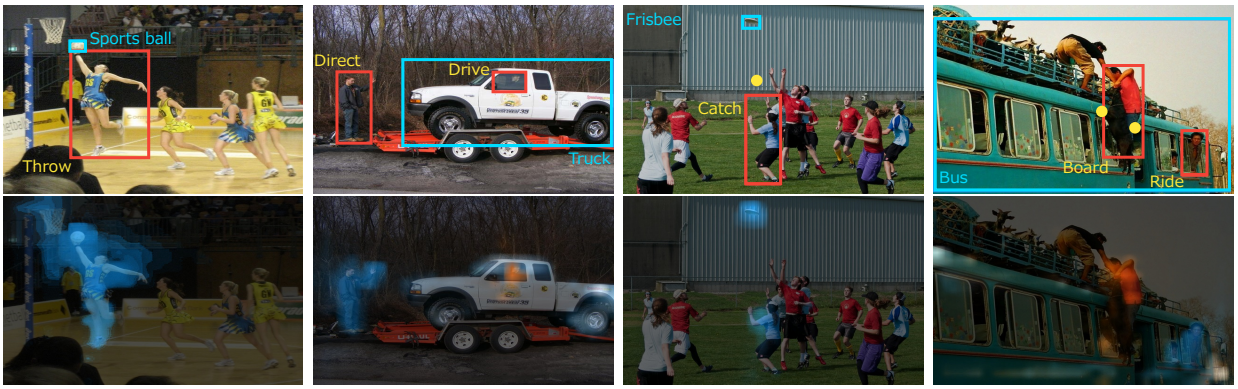
Table 1. Comparison against state-of-the-art methods on HICO-DET. The top, middle, and bottom blocks show the mAPs of the two-stage, single-stage, and our methods, respectively.

	Default			Known object		
Method	full	rare	non-rare	full	rare	non-rare
PMFNet [15]	17.46	15.65	18.00	20.34	17.47	21.20
Wang *et al*. [16]	17.57	16.85	17.78	21.00	20.74	21.08
In-GraphNet [18]	17.72	12.93	19.31	–	–	–
VSGNet [14]	19.80	16.05	20.91	–	–	–
FCMNet [11]	20.41	17.34	21.56	22.04	18.97	23.13
ACP [8]	20.59	15.92	21.98	–	–	–
PD-Net [19]	20.81	15.90	22.28	24.78	18.88	26.54
DJ-RM [9]	21.34	18.53	22.18	23.69	20.64	24.60
VCL [6]	23.63	17.21	25.55	25.98	19.12	28.03
ConsNet [12]	24.39	17.10	26.56	–	–	–
DRG [3]	24.53	19.47	26.04	27.98	23.11	29.43
UnionDet [7]	17.58	11.72	19.33	19.76	14.68	21.27
Wang *et al*. [17]	19.56	12.79	21.58	22.05	15.77	23.92
PPDM [10]	21.73	13.78	24.10	24.58	16.65	26.84
Ours (ResNet-50)	29.07	21.85	31.23	31.68	24.14	33.93
Ours (ResNet-101)	**29.90**	**23.92**	**31.69**	**32.38**	**26.06**	**34.27**

D. Computational Efficiency Comparison

To analyze the model efficiency of our QPIC, we compare the inference times of QPIC and PPDM [10], which is one of the highest speed models. We used the publicly available source code of PPDM[1], and tested each model on a single Tesla V100 GPU with CUDA ver. 10.1 and PyTorch

[1]https://github.com/YueLiao/PPDM



(a) (b) (c) (d)

Figure 1. Typical failure cases of conventional detectors (top row) and attentions of QPIC (bottom row). The ground-truth human bounding boxes, object bounding boxes, object classes, and action classes are drawn with red boxes, blue boxes, blue characters, and yellow characters, respectively. In (b) and (d), the attentions corresponding to different HOI instances are drawn with blue and orange, and the areas where two attentions overlap are drawn with white.

Table 2. Comparison against state-of-the-art methods on V-COCO. The split of the blocks are the same as Table 1.

Method	Scenario 1	Scenario 2
VCL [6]	48.3	–
In-GraphNet [18]	48.9	–
DRG [3]	51.0	–
VSGNet [14]	51.8	57.0
PMFNet [15]	52.0	–
PD-Net [19]	52.6	–
Wang *et al*. [16]	52.7	–
ACP [8]	53.0	–
FCMNet [11]	53.1	–
ConsNet [12]	53.2	–
UnionDet [7]	47.5	56.2
Wang *et al*. [17]	51.0	–
Ours (ResNet-50)	**58.8**	**61.0**
Ours (ResNet-101)	58.3	60.7

Table 3. Comparison of the efficiency.

Method	HICO-DET (mAP)	Inference time (ms)
PPDM [10]	21.73	64
Ours (ResNet-50)	29.07	**46**
Ours (ResNet-101)	**29.90**	63

ver. 1.5 [13]. Table 3 shows the comparison result. As the table shows, the inference time of QPIC with the ResNet-50 backbone is smaller by 18 ms than that of PPDM. In particular, PPDM takes 17 ms to organize outputs from the network, while QPIC takes only 5.4 ms to do that. These results indicate that QPIC is more efficient than conventional methods mainly because the simple detection heads of QPIC realize the simple inference procedures.

E. Additional Qualitative Analysis

Figure 1 shows the additional failure cases of conventional methods. Figure 1a and 1b show the failure cases of DRG [3], and Fig. 1c and 1d show those of PPDM [10], where QPIC successfully detects the human-object interactions (HOIs). As discussed in the main manuscript, the regions in an image other than a human and object bounding box sometimes contain useful information. Fig. 1a is a typical example case, where the basketball goal is likely to be the important contextual information. The attention of QPIC shows that it aggregates features from the region of the basketball goal, resulting in the correct detection. Figure 1b shows an example case where multiple HOI instances are overlapped. As shown in the figure, the bounding box of the track includes that of the driving human, which may induce contaminated features. The performance is degraded by this contamination. Unlike DRG, QPIC selectively aggregates features for each HOI using the attention mechanism as shown in the attention map, and successfully detects the HOIs. In Fig. 1c and 1d, the features of the detection points, which are the locations to predict HOIs in PPDM and drawn in the yellow circles in the figures, are likely to be dominated by irrelevant information because the points are on the background or irrelevant human. As is the case with DRG, PPDM cannot predict HOIs with these contaminated features, while QPIC can do it with the selectively aggregated features.

References

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

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

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

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

[5] Saurabh Gupta and Jitendra Malik. Visual semantic role labeling. May 2015. arXiv:1505.04474.

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

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

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

[9] 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*, June 2020.

[10] 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*, June 2020.

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

[12] Ye Liu, Junsong Yuan, and Chang Wen Chen. ConsNet: Learning consistency graph for zero-shot human-object interaction detection. In *ACM Multimedia*, October 2020.

[13] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In *NeurIPS*, December 2019.

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

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

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

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

[18] Dongming Yang and Yuexian Zou. A graph-based interactive reasoning for human-object interaction detection. In *IJCAI*, July 2020.

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