Supplementary Material for Object-Centric Multi-Task Learning for Human Instances

Hyeongseok Son
hs1.son@samsung.com
Sangil Jung
sang-il.jung@samsung.com
Solae Lee
solae913.lee@samsung.com
Seongeun Kim
se91.kim@samsung.com
Seung-In Park
si14.park@samsung.com
ByungIn Yoo
byungin.yoo@samsung.com

Samsung Advanced Institute of Technology (SAIT)
Suwon, Republic of Korea

1 Details About Network Architecture

1.1 Network architecture

As described in Sec. 3 in the main paper, our network consists of image feature extractor, transformer decoder, and task-specific modules. Our main difference in the architecture is to concentrate on the transformer decoder. The image feature extractor is based on the Mask2Former [1], which consists of a backbone network and a pixel decoder. We can select various backbone models such as residual network [2] and Swin-transformer [4]. Following [1], we use multi-scale deformable attention Transformer (MSDeformAttn) [7] as the pixel decoder. Regarding the transformer decoder, each individual human in the given image corresponds to each of proposed human-centric queries, and we use 100 queries for all experiments. The decoder consists of 8 decoder layers. Distinct form [1], in each decoder layer, we use a deformable attention layer [7] as a cross-attention layer, and a self-attention layer is placed before the deformable attention layer. Each deformable attention layer receives 3 scales of image feature ($1/32$, $1/16$, $1/8$ of the feature resolution) from the pixel decoder.

1.2 Multi-scale attention with keypoints

While, for simplicity in Eq. 6 in the main paper, we describe the mathematical derivation of deformable attention with keypoints at a single scale, the equation can naturally be used to the case of multi-scale attention module. Assume that there are N_s image feature maps $x_s \in \mathbb{R}^{H_s \times W_s \times D}$ for scale index s. By repeatedly applying the sampling process of Eq. 5 in the main paper for each x_s, sampled feature $V_{m,s}$ is obtained for each scale s. The multi-scale

attention value $V_m \in \mathbb{R}^{N_p N_s \times D/N_h}$ is defined as

$$V_m = Cat\left(V_{m,1}^T, \ldots, V_{m,N_s}^T\right)^T.$$

Now, the multi-scale attention output can be obtained by the same Eq. 6 in the main paper if one assumes the attention coefficient having wider dimension: $A_m \in \mathbb{R}^{1 \times N_p N_s}$.

2 Details About Experimental Setting

2.1 Loss function

We use a binary cross-entropy loss for human classification. For the segmentation, we use a binary cross-entropy and the dice loss [8] as was done in [1]. For the regression of the bounding box and joint positions, we use a RLE loss function [3] because RLE is known to be better than L_1 loss for regression problems. We train multiple tasks together with our unified architecture. To this end, a total training loss is defined with the weighted summation of multiple loss functions for the tasks. Regarding bipartite matching, we use the classification and the mask loss [1] instead of using all the task losses. We empirically found that it is sufficient for matching.

Specifically, we use four types of loss functions according to the target tasks: classification, segmentation, bbox, and pose. We denote these loss functions as L_c, L_s, L_b, and L_p, respectively. Then, the total training loss is defined as

$$L_{total} = \lambda_c L_c + \lambda_s L_s + \lambda_b L_b + \lambda_p L_p. \tag{1}$$

where λ_c, λ_s, λ_b and λ_p are the mixing weights and are set to 2, 5, 0.2, and 0.2, respectively, in our experiment. Because there are many possible combinations of mixing weights for multiple tasks, another best combination would exist. Still, our models with these weights show practically reasonable performance in the target tasks without sophisticated tuning to search the best hyper-parameters.

The training loss is applied to the matched pairs of instances between the prediction and the ground-truth. We also use an auxiliary training loss by attaching the prediction layer to each transformer decoder layer, similar to [1]. The auxiliary loss is same with L_{total}.

2.2 Data augmentation

As described in Sec. 4 in the main paper, we follow a data augmentation scheme used in [1]. Specifically, for each image, we apply random scaling to the image with the range [0.1, 2.0] and crop the scaled image with the fixed size of 1024×1024. If the scaled image is smaller than the cropping size, we apply zero-padding to right- and bottom-side of the image to produce the result image of 1024×1024.

3 Additional Experiments

3.1 Canonical space of the pose part of learnable keypoints

As described in Sec. 3.1 in the main paper, we normalized the coordinates of the *pose part* in our learnable keypoints by the box coordinate of the *bbox part*; we refer it as the canonical

Pose joint coordinates of learnable keypoints	Accuracy (mAP)		
	Det.	Pose.	Seg.
In the image space	54.8	55.3	49.6
In the canonical space	56.1	64.4	51.7

Table 1: Effectiveness of employing the canonical space for the pose part of our learnable keypoints.

Variants of Learnable keypoints	Accuracy (mAP)		
	Det.	Pose.	Seg.
Canonical space coordinate	56.1	64.4	51.7
Image space coordinate	56.2	63.0	51.5
Canonical space embedding	56.2	63.7	51.8

Table 2: Effect of different forms of leranable keypoints used in task-specific heads (segmentation and detection).

space. We can also define the coordinates of *pose part* in the image space instead of canonical space. We empirically found that representing the *pose part* in the image space causes a large performance drop in pose estimation and it also degrades the accuracy of two other tasks (Table 1).

3.2 Variants of the learnable keypoints

In the experiments in the main paper, our proposed components successfully improve learnable keypoints and the performance of various tasks. Meanwhile, different forms of learnable keypoints as conditional information may lead more performance improvement because there can be a suitable form of the conditional information for each task. In this experiment, we explore the possibility. In the previous experiment, we empirically found that the current form using the coordinates of pose joints in the canonical space is suitable for learning pose estimation. Therefore, we test different forms for only object detection and segmentation tasks. For the tasks, we test other forms such as the joint coordinates in the image space or keypoint embedding instead of coordinate. For keypoint embedding, we use the same process of obtaining structural embedding used in (Eq. 3 in the main paper).

The canonical space coordinate has better accuracy in pose estimation compared to the other variants while having the comparable accuracy on detection and segmentation tasks (Table 2). This implies that our coordinate information in learnable keypoints can be generally applied to various tasks.

3.3 OCHuman dataset

In this section, we present quantitative results on the OCHuman dataset [6]. Compared to Pose2Seg [6], our method achieves better performance in both detection and segmentation tasks (Table 3). Compared to a state-of-the-art segmentation method (Mask2Former [1]), our method shows a comparable accuracy in segmentation. Compared to our baseline multi-tasking model (BaseNet-DPS), our approach still induces significant improvements in pose estimation and segmentation on this different dataset.

We show additional visual results of our methods on the COCO and OCHuman datasets (Fig. 1).

Model	Backbone	OCHuman Val (mAP)			OCHuman Test (mAP)		
		Det.	Pose.	Seg.	Det.	Pose.	Seg.
Pose2Seg	R-50	✗	28.5	22.2	✗	30.3	23.8
Mask2Former	Swin-B	✗	✗	27.5	✗	✗	27.8
BaseNet-DPS	Swin-B	19.8	30.2	25.6	19.4	29.7	25.5
HCQNet	Swin-B	19.7	31.0	27.1	19.4	30.9	27.3

Table 3: Comparison on the OCHuman Dataset.

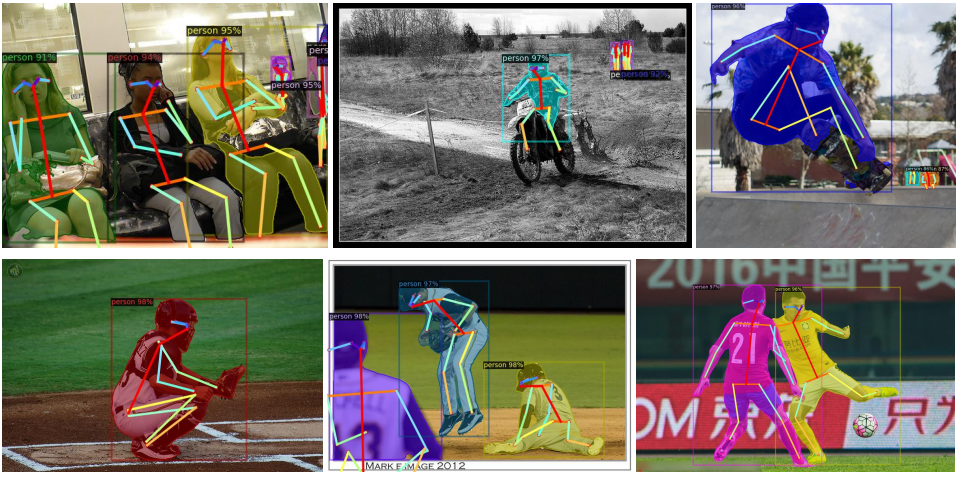

Figure 1: Additional visual results of our HCQNet on scenes containing dynamic human poses.

References

[1] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proc. CVPR*, 2022.

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

[3] Jiefeng Li, Siyuan Bian, Ailing Zeng, Can Wang, Bo Pang, Wentao Liu, and Cewu Lu. Human pose regression with residual log-likelihood estimation. In *Proc. ICCV*, 2021.

[4] 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 *Proc. ICCV*, 2021.

[5] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-net: Fully convolutional neural networks for volumetric medical image segmentation. In *Proc. 3DV*, 2016.

[6] Song-Hai Zhang, Ruilong Li, Xin Dong, Paul Rosin, Zixi Cai, Xi Han, Dingcheng Yang, Haozhi Huang, and Shi-Min Hu. Pose2seg: Detection free human instance segmentation. In *Proc. CVPR*, 2019.

[7] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: deformable transformers for end-to-end object detection. In *Proc. ICLR*, 2021.