Instance-Adaptive and Geometric-Aware Keypoint Learning for Category-Level 6D Object Pose Estimation

Supplementary Material

1. More implementation details

Here we offer more implementation details about our AG-Pose. For the Feature Extractor, we use a PSP Network [8] based on ResNet-18 [1] to extract the image feature and implement the PointNet++ [3] with 4 set abstract layers and multi-scale grouping to extract the point feature. For the IAKD, the attention layers contains of four attention blocks, and each of them is a standard attention operation [5] with $d_{model} = 256$ and $num\_heads = 4$. For other training details, we train our network for 50 epochs with a total of 200,000 iterations. The batch size is set as 24, with a ratio of 3:1 between real and synthetic data. All experiments are conducted on a single RTX 3090 GPU with an Intel Xeon Gold 6248R @ 4.000 GHz CPU. We implement our code using PyTorch 1.12.1 and CUDA 11.3.

2. Symmetry handling

The issue of object symmetry is common in pose estimation. To handle it, following [4], we map ambiguous rotations to a canonical one. Since all symmetric objects in NOCS datasets [6] exhibit continuous symmetry along the y-axis, for a rotation R, the above process is as follows,

$$Map(R) = R\hat{S}, \tag{1}$$

$$\hat{S} = \begin{bmatrix} cos\hat{\theta} & 0 & -sin\hat{\theta} \\ 0 & 1 & 0 \\ sin\hat{\theta} & 0 & cos\hat{\theta} \end{bmatrix}, \tag{2}$$

$$\hat{\theta} = arctan2(R_{13} - R_{31}, R_{11} + R_{33}). \tag{3}$$

3. Accuracy, memory usage, total parameters and inference speed of models with different N_{kpt}

To demonstrate the efficiency and scalability of our AG-Pose, we show the quantitative results of the accuracy, GPU memory usage during training, total parameters and inference speed of our model with different number of keypoints N_{kpt} in Table 1. As demonstrated by the results, the accuracy of our AG-Pose improves when increasing N_{kpt}. It is worth noting that the increment of the GPU memory usage, the total parameters and the computational overhead of our model are slight and acceptable. Specifically, when we lift the number of keypoints N_{kpt} from 16 to 128, the total parameters of the model just increase by less than 1%, and the increase in GPU memory usage is also affordable. Additional, the inference speed of our model only decreases by less than 10%. We attribute this to that: 1) The number of keypoints N_{kpt} is quite small compared to the size of point cloud N. Using this sparse set of keypoints to represent the shapes of objects is effective and efficient. 2) The two-stage feature aggregation pipeline in proposed GAFA module is efficient, which can inject local and global geometric information into keypoints with a low computational burden.

4. Per-category results

The NOCS datasets [6] contains six different categories: bottle, bowl, camera, can, laptop and mug. We show the per-category and the average results of our AG-Pose on the REAL275 and CAMERA25 datasets in Table 2 and Table 3, respectively. It is worth noting that we train a single model for all categories as [2, 7, 9].

5. Keypoint heatmaps visualization

Here we visualize the query-instance heatmap $\mathbf{H}$ in the proposed Instance-Adaptive Keypoint Detector in Figure 1. We use the bottle category as an example. Specifically, we draw the heatmaps for the same query across different instances as well as heatmaps for different queries on the same instance. As shown in the Figure, different queries can focus on different parts of the input instance to comprehensively depict the shape of the object. On the other hand, the same query tends to focus on regions with similar structures across different instances, which demonstrates the generalizability of our IAKD within the specific category.

6. More visualization on REAL275

Here we visualize more pose predictions of our AG-Pose in Figure 2. In particular, we choose four images per scene for all six unseen scenes in the REAL275 validation set, in which red/green indicates the predicted/gt results.

Table 1. **Comparisons between models with different N_{kpt}.**

N_{kpt}	IoU_{75}	5° 2 **cm**	Memory	Parameters	Speed(FPS)
16	78	47.9	11.6G	207,529,575	30.61
32	78.3	48.8	12.1G	207,541,863	30.33
64	**79.7**	51	13.4G	207,566,439	29.88
96	79.5	**54.7**	14.4G	207,591,015	27.98
128	78.8	52.8	16.4G	207,615,591	27.58

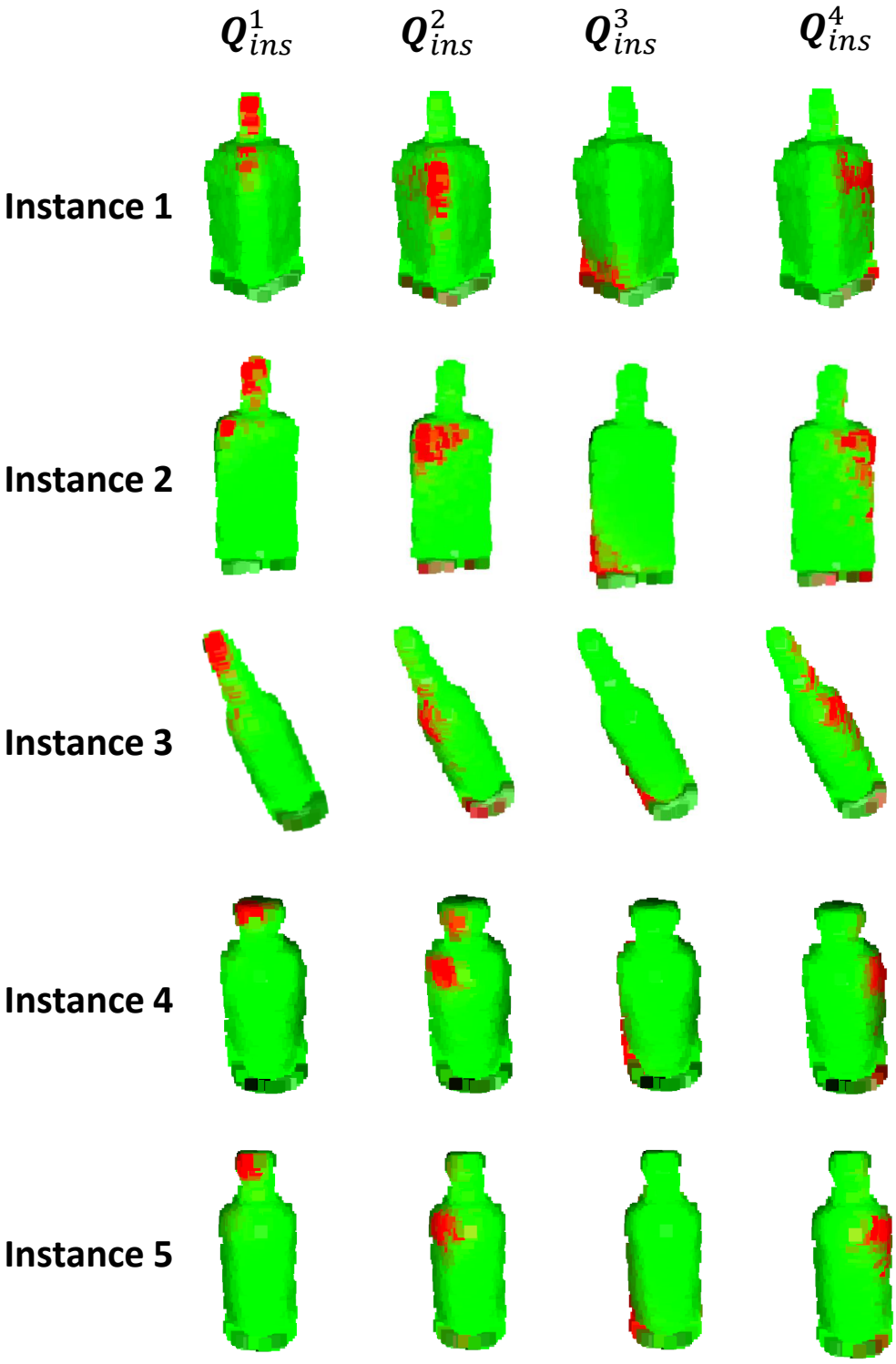

Figure 1. **Query-instance heatmaps in the proposed IAKD.** Each row represents the heatmaps for different queries on a same instance. Each column represents the heatmaps for a same query across different input instances. Red/green indicates a large/small weight.

References

[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 770–778, 2016. 1

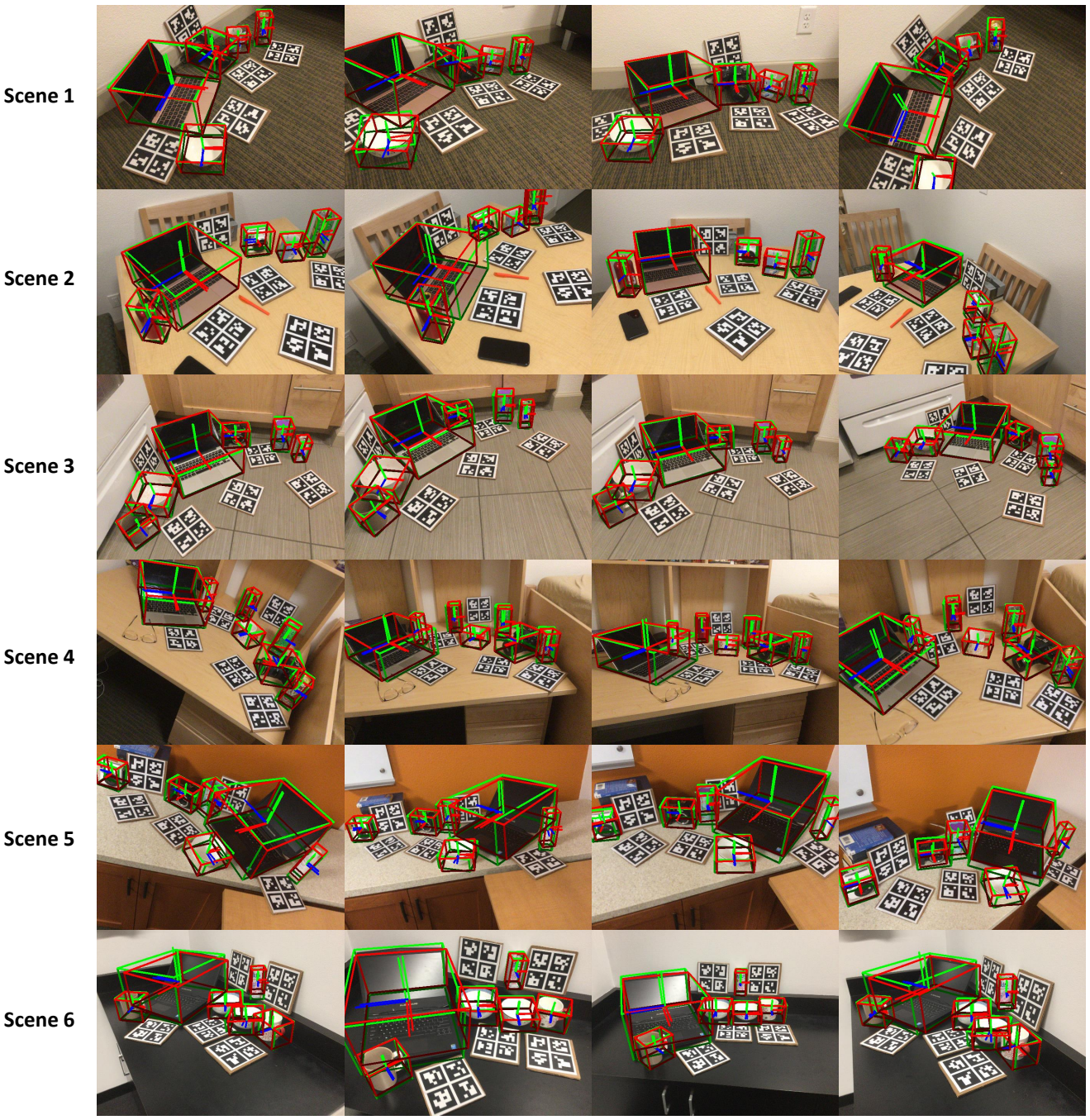

Figure 2. **More qualitative results of our AG-Pose on the REAL275 dataset.** Red/Green indicates the predicted/gt results.

[2] Jiehong Lin, Zewei Wei, Changxing Ding, and Kui Jia. Category-level 6d object pose and size estimation using self-supervised deep prior deformation networks. In European Conference on Computer Vision, pages 19–34. Springer, 2022. 1

[3] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In Advances in Neural Information Processing Systems (NeurIPS), pages 5099–5108, 2017. 1

[4] Meng Tian, Marcelo H Ang Jr, and Gim Hee Lee. Shape prior deformation for categorical 6d object pose and size estimation. In Proceedings of the European Conference on Computer Vision (ECCV), 2020. 1

[5] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. Advances in neural information processing systems, 30, 2017. 1

Table 2. **Per-category results of AG-Pose on REAL275.**

category	IoU_{50}	IoU_{75}	$5°\,2\,$**cm**	$5°\,5\,$**cm**	$10°\,2\,$**cm**	$10°\,5\,$**cm**
bottle	57.7	50.3	62	64.9	83.4	88
bowl	100	100	88.7	94.3	94.1	99.7
camera	90.8	82.9	1.2	1.3	24.8	27.3
can	71.3	71.2	83.4	85.3	96.3	98.6
laptop	83.3	74.1	59.6	91.1	61.2	95.6
mug	99.4	98.5	32.9	33.4	88.3	89.3
average	83.7	79.5	54.7	61.7	74.7	83.1

Table 3. **Per-category results of AG-propose on CAMERA25.**

category	IoU_{50}	IoU_{75}	$5°\,2\,$**cm**	$5°\,5\,$**cm**	$10°\,2\,$**cm**	$10°\,5\,$**cm**
bottle	93.7	91.4	80.9	96.4	82.3	99
bowl	96.9	96.7	98.7	99	99.7	99.8
camera	89.2	84.3	57	60.9	73.6	81.1
can	92.1	92	99.7	99.8	99.7	99.9
laptop	97.5	90.8	76.1	85.9	80.6	92.4
mug	93.6	92.7	54.5	54.6	77.1	77.3
average	93.8	91.3	77.8	82.8	85.5	91.6

[6] He Wang, Srinath Sridhar, Jingwei Huang, Julien Valentin, Shuran Song, and Leonidas J. Guibas. Normalized object coordinate space for category-level 6d object pose and size estimation. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2019. 1

[7] Ruiqi Wang, Xinggang Wang, Te Li, Rong Yang, Minhong Wan, and Wenyu Liu. Query6dof: Learning sparse queries as implicit shape prior for category-level 6dof pose estimation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 14055–14064, 2023. 1

[8] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 2881–2890, 2017. 1

[9] Linfang Zheng, Chen Wang, Yinghan Sun, Esha Dasgupta, Hua Chen, Aleš Leonardis, Wei Zhang, and Hyung Jin Chang. Hs-pose: Hybrid scope feature extraction for category-level object pose estimation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 17163–17173, 2023. 1