Supplementary material for GeoReF: Geometric Alignment Across Shape Variation for Category-level Object Pose Refinement

Linfang Zheng[1,3] Tze Ho Elden Tse[*3] Chen Wang[*1,2] Yinghan Sun[1] Hua Chen[1]
Aleš Leonardis[3] Wei Zhang[†1] Hyung Jin Chang[3]

[1]Shenzhen Key Laboratory of Control Theory and Intelligent Systems, School of System Design and Intelligent Manufacturing, Southern University of Science and Technology, China
[2]Department of Computer Science, the University of Hong Kong, China
[3]School of Computer Science, University of Birmingham, UK

{lxz948, txt994}@student.bham.ac.uk, cwang5@cs.hku.hk, sunyh2021@mail.sustech.edu.cn
{chenh6,zhangw3}@sustech.edu.cn,{a.leonadis,h.j.chang}@bham.ac.uk

1. About the Runtime

On a machine with an Intel 13900k CPU and a Nvidia RTX 4090 GPU, the speed of our proposed method is 67.5 FPS for 1 iteration, and 22.3 FPS when using 4 iterations.

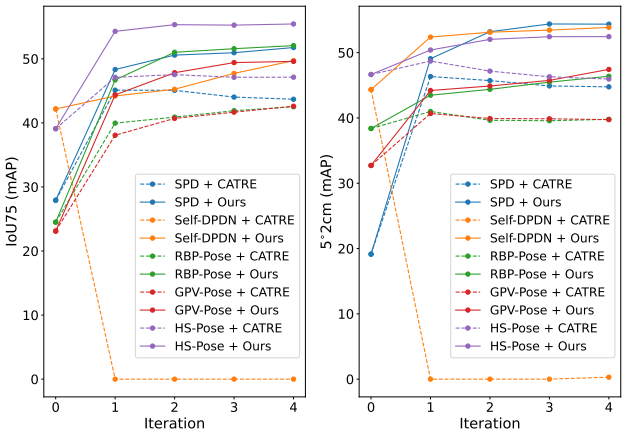
2. Effect of number of iterations

We find that the performance of our proposed method saturates after 4 iterations. Therefore, we set the iteration number to 4 for our experiments. We provide a line graph to show the performance changes of our method and CATRE [6] during the iteration in Fig. 1 We show that our proposed method consistently outperforms the baseline method and saturates after 4 iterations in both figures.

3. Ablation Studies

Refinement with different initial estimations. Apart from the table provided in the main paper, we visually show the robustness of our method on different initial estimations generated from 5 pose estimation methods [2, 5, 8, 11, 13] with ranging performance. As shown in Fig. 1, our method keeps improving the performance of the initial estimations, while CATRE [6] failed when refining the initial estimations from Self-DPDN [5]. Additionally, our method keeps improving during the iterations, while CATRE's performance starts to decrease after one iteration (see the dashed lines in Fig. 1).

The effect of CCT. To demonstrate the effect of CCT, we show a statistics plot of feature distances before and after CCT on objects with different shape complexities of the CAMERA25 test set. In this experiment, the initial pose of the shape prior is aligned with the ground truth pose to guarantee that the observed variations in feature distance are solely attributable to differences in shape. As shown in Fig 2, the feature distance between the shape prior and the input target shrinks significantly after applying CCT.



(a) (b)

Figure 1. **Comparison between CATRE and our method on different initial estimations across different refining iterations.** (a) IoU_{75} performance comparison. (b) $5°2\text{cm}$ performance comparison. Our methods are shown in solid lines and CATRE's are in dashed lines. Iteration 0 shows the performance of the initial estimations.

4. Generalizability test on CAMERA25

More Results. To test the generalizability of our method when trained on a small dataset and tested on a large dataset, we randomly sample datasets from the CAMERA25 train-

[*]Equal contribution, order by dice rolling.
[†]The corresponding author.

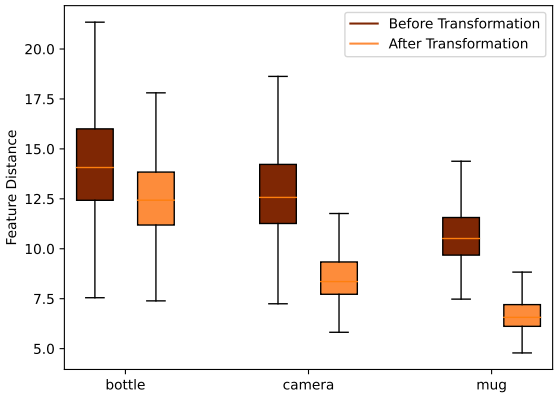


Figure 2. **Feature distances between the shape prior and the input point cloud before and after applying the cross-cloud transformation.**

ing dataset at different ratios (2%, 4%, and 6%). This yields training sizes of 5k, 10k, 15k. We show the results of the generalizability test in Table 1. We observe that our method, trained only using 2% of the train set, can already outperform a fully trained CATRE on all training data. Also, our performance becomes stable when using 4% of the train set (see Table 1 [C1, D1]), while CATRE requires additional training data for better performance. Since our performance became stable, we did not test on larger data sizes.

Experiment settings: To ensure the distribution of different categories in the sampled mini datasets, we control the image number of each object in the sampled datasets: 1) 5 images per object for the 2% train set, 2) 10 images for the 4% train set, and 3) 15 images for the 6% train set.

Table 1. **The generalizability test on the CAMERA25 dataset.** Higher score means better performance. Overall best results are in bold, and the second-best results are underlined. The training data size is denoted as *T. Size*.

Row	Method	T. Size	IoU_{75}	5°2cm	5°5cm	10°2cm	10°5cm
A0	CATRE	275k	76.1	75.4	80.3	83.3	89.3
B0	CATRE	5k	63.2	66.4	72.3	79.4	87.4
B1	Ours	5k	77.5	75.4	81.1	83.4	<u>90.0</u>
C0	CATRE	10k	66.5	69.7	75.5	81.8	89.1
C1	Ours	10k	**79.2**	<u>77.9</u>	<u>84.0</u>	**83.8**	**90.5**
D0	CATRE	15k	69.7	73.2	78.8	82.6	89.4
D1	Ours	15k	<u>78.1</u>	**78.0**	**84.1**	<u>83.6</u>	**90.5**

5. Detailed Network Architectures

The network structure of the HS Feature Extractor and the Pose Error Predictor is shown in Fig. 2 of the main paper. The structure of the Pose Error Predictor for $\Delta\mathbf{R}$ estimation and the $\Delta\mathbf{t}, \Delta\mathbf{s}$ estimation are identical, we follow the CATRE [6] and use 3 Convolution-1D layers with permutation before the final layer to generate the pose errors. For the Matrix Net, we follow PointNet [7] first use 3 Convolution-1D layers with $[64, 128, 1024]$ output dimensions and a kernel size of 1 to extract the dense point features, then the features going through a maximum pooling layer and 3 liner layers with [512, 256, $\boldsymbol{f}_{\text{LAT}}$] to generate the matrix. For the first Matrix Net that generates the adaptive affine transformation (LAT) for the input point cloud, $\boldsymbol{f}_{\text{LAT}}$ is 9. For the second Matrix Net, $\boldsymbol{f}_{\text{LAT}}$ is 8192, as it outputs two LATs with the matrix size of $\mathbb{R}^{64\times64}$. In the final structure of the GeoReF, we use two HS-layers to replace the first two Convolution-1D layers in the second Matrix Net, which in our experiments, show slightly better results than without HS-layers (See Table 2 [B0, G0] for the performance comparison). The structure of the Global Feature Extractor is shown in Fig. 3, we use 1 layer of HS-layer and 2 Convolution-1D layers with the output size of $[128, 512, 1024]$ to extract dense point features, and then apply maximum pooling to get the global feature. Finally, the global feature is concatenated with the input features for the outputs.

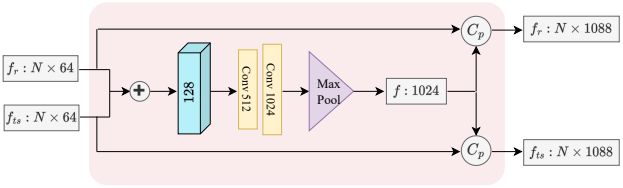


Figure 3. **Structure of the global extractor.**

6. Performance Comparion on CAMERA25

Table 3 compares the accuracy of our method with the state-of-the-arts. As discussed in Sec. 4, our performance stabilizes when using 4% of the full train set. Therefore, we present the results obtained with this training size. As shown in Table 3, we greatly enhanced the performance of SPD, resulting in a performance that outperformed state-of-the-art pose estimation methods. Specifically, we improved the performance of SPD [8] on IoU_{75} by 32.7%, 5°5cm by 25.2%, and 5°2cm by 23.8%. We also outperform the baseline CATRE on IoU_{75} by 3.1%, 5°5cm by 3.7%, and 5°2cm by 2.5%. Additionally, we show our results trained using 5k images (2%) of the train set, which already outperforms the state-of-the-art methods.

7. Per-category Performance

7.1. CAMERA25.

We present our per-category object pose refinement performance in Table 4. We use SPD [8] as the initial estimation method and report the performance after 4 refinement iterations. We show that our method largely improved the initial performance.

Table 2. **Ablation studies on REAL275.**
Higher score means better performance. Overall best results are in bold. Row's code in bold means the strategies taken in the final structure.

Row	Method	IoU_{50}	IoU_{75}	5°2cm	5°5cm	10°2cm	10°5cm	2cm	5°
A0	CATRE [6] (baseline)	77.0	43.6	45.8	54.4	61.4	73.1	75.1	58.0
B0	**Ours**: E0+Cross-Cloud Transformation	79.2 **2.2↑**	**51.8** **8.2↑**	**54.4** **8.6↑**	**60.3** **5.9↑**	**71.9** **10.5↑**	**79.4** **6.3↑**	**81.9** **6.8↑**	**64.3** **6.3↑**
C0	A0: PointNet → HS-Encoder	71.0	30.1	41.9	45.9	60.6	70.3	71.9	48.7
C1	A0: PointNet → 3DGCN-Encoder	-	28.4	36.0	43.4	-	-	68.0	47.7
D0	A0 + prior in ST branch	77.1	45.8	48.0	54.6	63.8	72.5	77.9	59.2
E0	D0: PointNet → HS-layer+LATs	79.4	51.0	52.4	58.6	69.4	77.7	80.4	62.4
E1	B0: No LAT on input points	76.1	39.3	46.6	53.0	65.4	74.8	78.0	58.2
E2	B0: No LATs on features	78.5	48.8	47.4	53.0	67.4	75.0	80.4	57.4
E3	B0: No LAT on the rotation feature	**79.8**	50.6	50.4	56.2	68.6	76.3	80.2	60.8
F0	E0+ Global Concatenation Fusion	77.7	48.4	47.8	54.5	67.1	75.2	80.1	59.4
G0	B0: No HS-layer in Matrix Net	77.8	50.2	54.1	60.1	70.5	78.0	81.2	63.6

Table 3. **Comparison with other methods on CAMERA25.**
Higher score means better performance. Overall best results are in bold. SPD* is the implementation results from CATRE, which is similar to the original SPD results.

Method	IoU_{75}	5°2cm	5°5cm	10°2cm	10°5cm
NOCS [9]	37.0	32.3	40.9	48.2	64.6
DualPoseNet [4]	71.7	64.7	70.7	77.2	84.7
CR-Net [10]	75.0	72.0	76.4	81.0	87.7
SGPA [1]	69.1	70.7	74.5	82.7	88.4
SAR-Net [3]	62.6	66.7	70.9	75.3	80.3
SSP-Pose [12]	-	64.7	75.5	-	87.4
RBP-Pose [11]	-	73.5	79.6	82.1	89.5
GPV-Pose [2]	-	72.1	79.1	-	89.0
HS-Pose [13]	-	73.3	80.5	80.4	89.4
SPD* [8]	46.9	54.1	58.8	73.9	82.1
SPD*+CATRE [6]	76.1	75.4	80.3	83.3	89.3
SPD*+**Ours** (2%)	77.5	75.4	81.1	83.4	90.0
SPD*+**Ours**	**79.2**	**77.9**	**84.0**	**83.8**	**90.5**

7.2. REAL275.

We present the per-category object pose refinement results in Table 5. We use SPD [8] as the initial estimation method and report the performance after 4 refinement iterations. We show that our method largely improved the initial performance.

8. Additional Qualitative Results

We show additional qualitative results of our method test on different REAL275 test scenes in Fig. 4 and Fig. 5. We highlight the performance differences with red arrows.

Table 4. **Per-category results of our method on CAMERA25 dataset.**

Method	Category	IoU_{50}	IoU_{75}	$5°2cm$	$5°5cm$	$10°2cm$	$10°5cm$	$10°10cm$	$5°$	2cm
SPD	bottle	88.9	64.5	63.8	82.8	69.2	92.4	97.3	86.8	69.8
SPD+**Ours**	bottle	89.4	73.8	73.8	94.2	74.2	95.1	99.4	98.4	74.2
SPD	bowl	95.9	80.6	83.4	83.7	95.8	96.3	96.3	83.7	99.2
SPD+**Ours**	bowl	96.0	94.7	97.9	98.2	99.5	99.8	99.8	98.2	99.6
SPD	camera	61.9	4.7	27.3	29.3	72.9	78.6	78.6	29.5	89.8
SPD+**Ours**	camera	81.6	67.7	83.1	87.2	90.8	95.2	95.2	87.2	93.9
SPD	can	90.2	87.2	98.1	98.2	99.4	99.6	99.6	98.2	99.6
SPD+**Ours**	can	90.3	89.8	99.9	100.0	99.9	100.0	100.0	100.0	99.9
SPD	laptop	93.3	17.7	35.0	41.9	61.0	80.5	84.5	43.7	65.5
SPD+**Ours**	laptop	95.3	81.3	74.0	85.5	77.4	91.8	95.8	89.1	77.9
SPD	mug	82.7	24.1	15.5	15.5	44.1	44.1	44.1	15.9	99.6
SPD+**Ours**	mug	89.8	67.7	39.0	39.0	61.0	61.0	61.0	39.4	99.9

Table 5. **Per-category results of our method on REAL275 dataset.**

Method	Category	IoU_{50}	IoU_{75}	$5°2cm$	$5°5cm$	$10°2cm$	$10°5cm$	$10°10cm$	$5°$	2cm
SPD	bottle	49.9	13.1	21.6	23.2	69.4	76.0	87.1	35.9	80.7
SPD+**Ours**	bottle	49.8	36.2	64.8	68.0	82.5	88.6	100.0	82.5	89.1
SPD	bowl	100.0	77.1	50.5	54.0	75.8	80.3	80.3	54.0	94.7
SPD+**Ours**	bowl	100.0	91.9	91.2	95.6	95.4	100.0	100.0	95.7	95.4
SPD	camera	43.4	3.4	0.0	0.0	0.2	0.2	0.2	0.0	34.8
SPD+**Ours**	camera	78.4	12.4	2.1	2.1	17.9	18.8	18.9	2.2	58.3
SPD	can	70.0	29.8	37.9	42.7	80.4	91.6	91.6	45.5	87.1
SPD+**Ours**	can	70.3	36.7	75.6	78.6	96.0	99.9	99.9	80.7	96.0
SPD	laptop	82.0	35.5	4.6	7.0	24.5	65.3	65.9	7.1	29.1
SPD+**Ours**	laptop	80.8	73.9	67.6	91.8	68.9	94.4	95.6	92.5	69.3
SPD	mug	66.5	8.7	0.3	0.3	10.3	10.4	10.4	0.3	85.2
SPD+**Ours**	mug	96.2	59.5	24.8	25.9	70.7	74.8	74.8	25.9	89.9

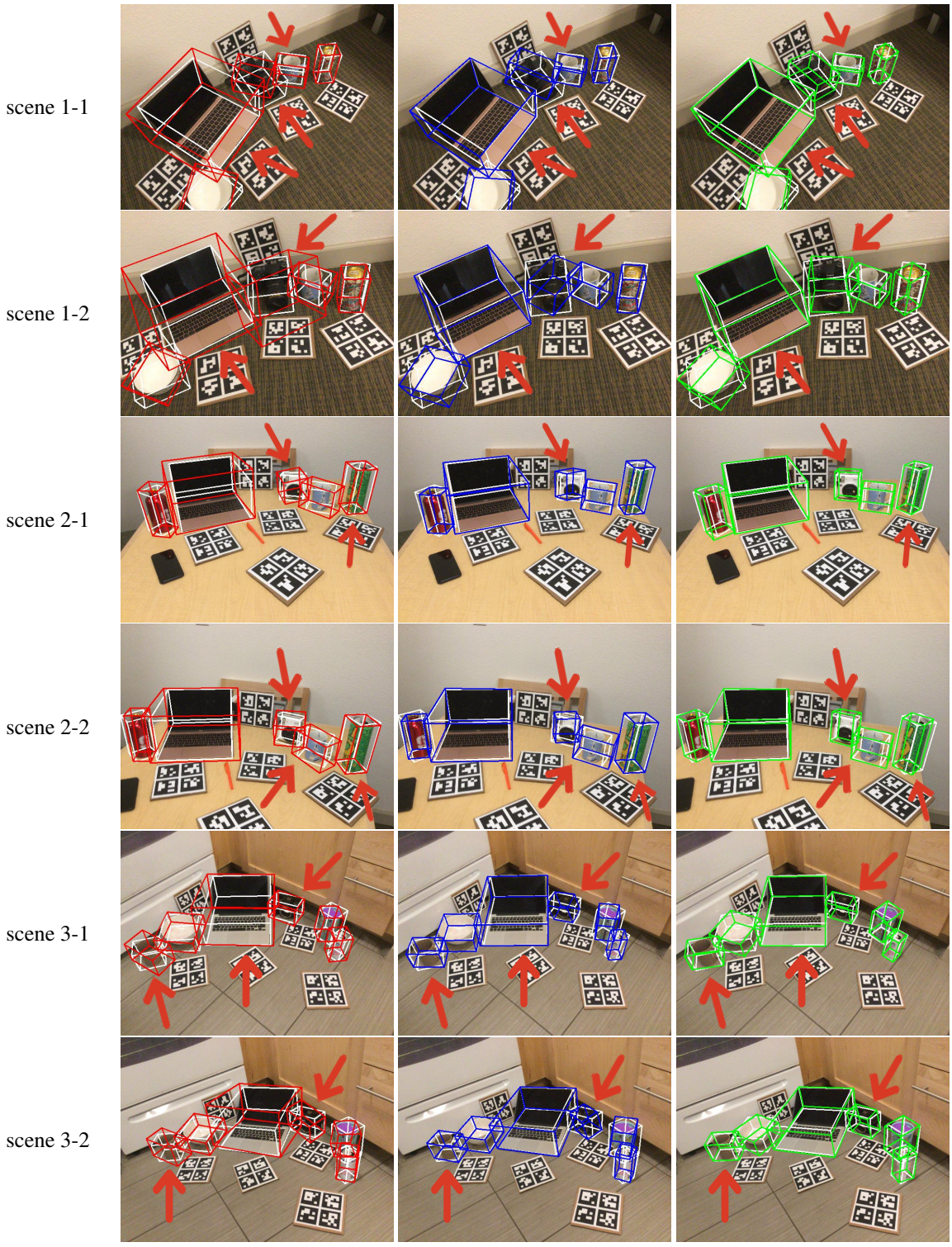


Figure 4. **More qualitative comparison between the proposed method (column #3) and the baseline method (column #2) use the SPD (column #1) as the initial estimation.** We choose two instances from each scene in REAL275 dataset. We show the ground truth with white lines. Note that the estimated rotations of symmetric objects (*e.g.* bowl, bottle, and can) are considered correct if the symmetry axis is aligned.

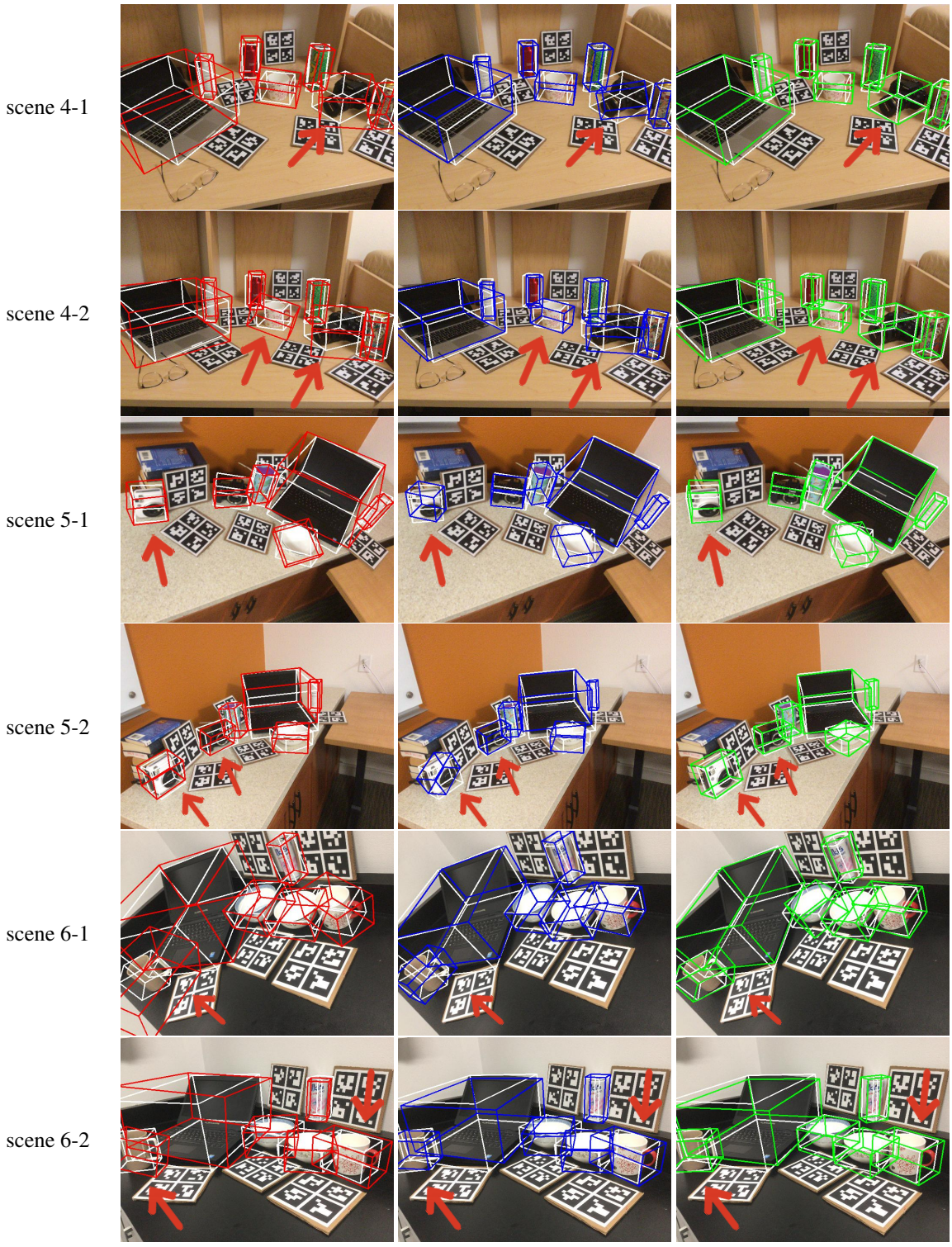


Figure 5. **More qualitative comparison between the proposed method (column #3) and the baseline method (column #2) use the SPD (column #1) as the initial estimation.** We choose two instances from each scene in REAL275 dataset. We show the ground truth with white lines. Note that the estimated rotations of symmetric objects (*e.g.* bowl, bottle, and can) are considered correct if the symmetry axis is aligned.

References

[1] Kai Chen and Qi Dou. SGPA: Structure-guided prior adaptation for category-level 6d object pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2773–2782, 2021.

[2] Yan Di, Ruida Zhang, Zhiqiang Lou, Fabian Manhardt, Xiangyang Ji, Nassir Navab, and Federico Tombari. GPV-Pose: Category-level object pose estimation via geometry-guided point-wise voting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6781–6791, 2022.

[3] Haitao Lin, Zichang Liu, Chilam Cheang, Yanwei Fu, Guodong Guo, and Xiangyang Xue. Sar-net: Shape alignment and recovery network for category-level 6d object pose and size estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6707–6717, 2022.

[4] Jiehong Lin, Zewei Wei, Zhihao Li, Songcen Xu, Kui Jia, and Yuanqing Li. DualPoseNet: Category-level 6d object pose and size estimation using dual pose network with refined learning of pose consistency. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3560–3569, 2021.

[5] Jiehong Lin, Zewei Wei, Changxing Ding, and Kui Jia. Category-level 6d object pose and size estimation using self-supervised deep prior deformation networks, 2022.

[6] Xingyu Liu, Gu Wang, Yi Li, and Xiangyang Ji. CATRE: Iterative point clouds alignment for category-level object pose refinement. In *European Conference on Computer Vision (ECCV)*, pages 499–516. Springer, 2022.

[7] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[8] Meng Tian, Marcelo H Ang, and Gim Hee Lee. Shape prior deformation for categorical 6d object pose and size estimation. In *European Conference on Computer Vision (ECCV)*, pages 530–546. Springer, 2020.

[9] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2642–2651, 2019.

[10] Jiaze Wang, Kai Chen, and Qi Dou. Category-level 6d object pose estimation via cascaded relation and recurrent reconstruction networks. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4807–4814. IEEE, 2021.

[11] Ruida Zhang, Yan Di, Zhiqiang Lou, Fabian Manhardt, Federico Tombari, and Xiangyang Ji. RBP-Pose: Residual bounding box projection for category-level pose estimation, 2022.

[12] Ruida Zhang, Yan Di, Fabian Manhardt, Federico Tombari, and Xiangyang Ji. SSP-Pose: Symmetry-aware shape prior deformation for direct category-level object pose estimation, 2022.

[13] 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.