Structure-Aware Correspondence Learning for Relative Pose Estimation

Supplementary Material

1. Results Across Angular Variations

To evaluate the performance of our method under significant angular variations, we choose the Objaverse dataset [2], as it provides a wide range of angular differences, making it particularly suitable for validating pose estimation accuracy. Following prior works [9, 10], we categorized the geodesic distances between reference and query poses into multiple groups, where each group represents increasing levels of angular difficulty. As the geodesic distance increases, the task becomes progressively more challenging, providing a rigorous benchmark to test the robustness of our approach under extreme conditions.

Figure 1 presents a performance comparison of our method against DVMNet [10] and 3DAHV [9]. Additionally, the 3DAHV [9] paper provides comparisons with a broader range of approaches, including 2D correspondence-based and hypothesis-and-verification-based methods. The combined results highlight the superior performance of our method, particularly under significant pose variations. As angular differences grow, our method's advantage becomes increasingly evident—while existing methods suffer notable accuracy drops, our approach maintains nearly 90% accuracy even at large angular ranges. This performance aligns with our motivation to effectively tackle the challenges posed by substantial viewpoint differences, demonstrating superior robustness as the angular variation increases.

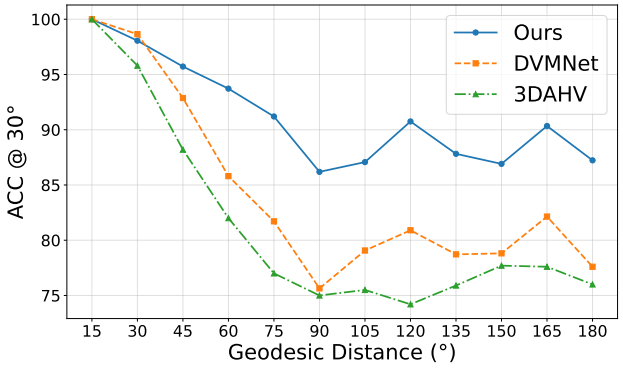

Figure 1. Acc @ 30° for different levels of object pose variation between the reference and query images, measured by geodesic distance.

2. Per-category Results

The CO3D dataset [6] includes 10 distinct categories. We present per-category results for these categories, as well as the overall average performance on the CO3D dataset. Additionally, we provide per-category results for the LINEMOD dataset [3]. These results are summarized in Tables 1 and 2, respectively.

Table 1. Per-category results on CO3D.

Category	mAE ↓	Acc@30° ↑	Acc@15° ↑
Ball	20.26	88.19	68.33
Book	11.80	96.39	88.67
Couch	13.63	93.60	81.80
Frisbee	16.08	88.00	74.40
Hot Dog	16.56	92.86	66.43
Kite	13.56	93.08	82.31
Remote	7.55	100.0	90.40
Sandwich	11.59	94.50	78.50
Skateboard	19.25	93.33	83.33
Suitcase	11.31	95.80	87.80
Mean	14.2	93.6	80.2

Table 2. Per-category results on LineMOD.

Category	mAE ↓	Acc@30° ↑	Acc@15° ↑
Cat	32.83	61.90	30.10
Bench Vise	14.19	96.60	62.40
Cam	30.78	72.60	33.20
Driller	18.77	90.50	53.40
Duck	39.49	59.00	29.80
Mean	27.2	76.2	41.8

3. Details of the Reconstruction Process

The extracted keypoints, represented by their spatial coordinates $\mathbf{X}_{kpt,q} \in \mathbb{R}^{N_{\text{kpt}} \times 2}$ and corresponding features $\mathbf{F}_{kpt,q} \in \mathbb{R}^{N_{\text{kpt}} \times C}$, serve as the input to the image reconstruction process. This section explains how these keypoints are utilized to reconstruct the query image $\mathbf{I}_q$.

3.1. Keypoint Feature Aggregation

To project the keypoint features $\mathbf{F}_{kpt,q}$ back into a dense spatial feature map $\mathbf{F}_{\text{recon}} \in \mathbb{R}^{C \times H \times W}$, inspired by previous works [4, 5], we use Gaussian heatmaps $\mathbf{G}_k(i,j)$ centered at each keypoint (x_k, y_k). The Gaussian heatmaps are defined as:

$$\mathbf{G}_k(i,j) = \exp\left(-\frac{(i-x_k)^2 + (j-y_k)^2}{2\sigma^2}\right), \quad (1)$$

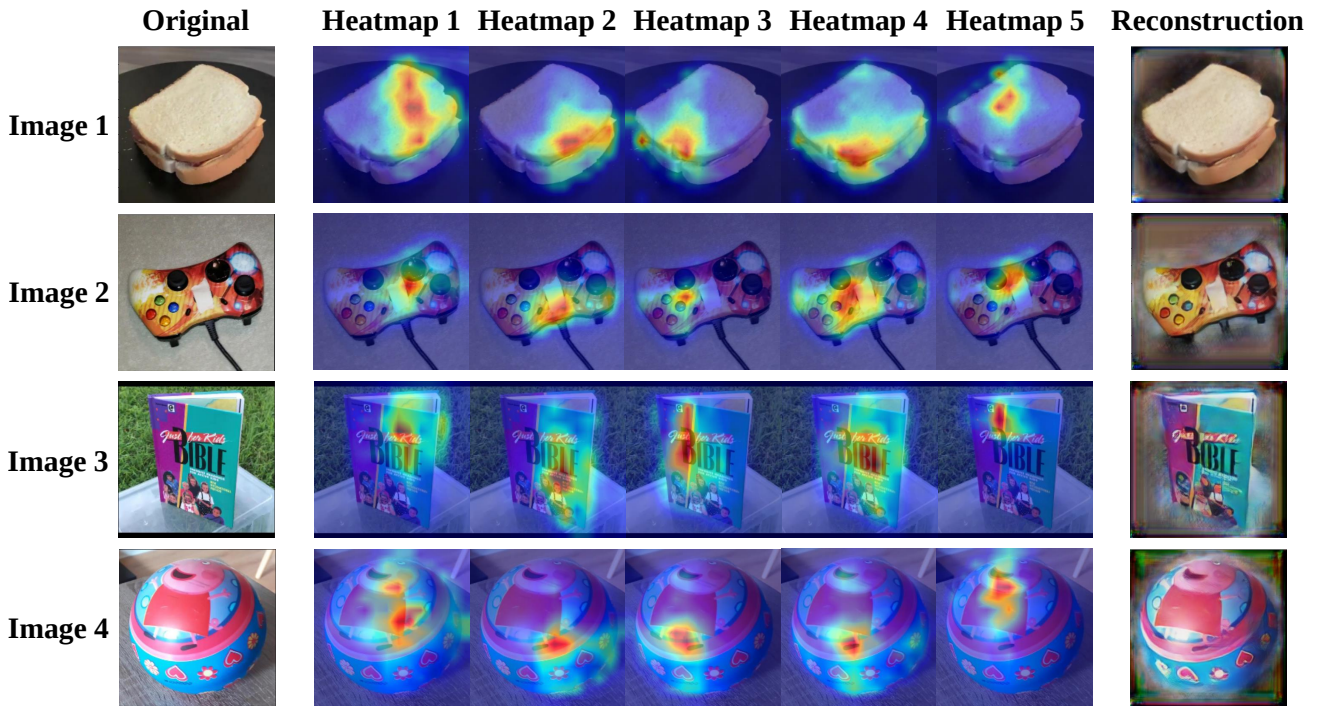

Figure 2. Visualization of keypoint heatmaps and reconstructed images. The first column shows the original input images, followed by five heatmaps representing the responses of the image-specific keypoint detectors for each image. The final column shows the reconstructed images.

where σ controls the spread of each keypoint's influence. The aggregated spatial feature at each pixel (i, j) is computed as:

$$\mathbf{F}_{\text{recon}}(i,j) = \sum_{k=1}^{N_{\text{kpt}}} \mathbf{G}_k(i,j) \cdot \mathbf{F}_{kpt,q,k} + \left(1 - \sum_{k=1}^{N_{\text{kpt}}} \mathbf{G}_k(i,j)\right) \cdot \bar{\mathbf{F}}_{kpt,q}, \tag{2}$$

where $\bar{\mathbf{F}}_{kpt,q}$ is the mean feature vector across all keypoints, providing a global context for regions not directly influenced by any specific keypoint.

3.2. Image Reconstruction from Features

The aggregated spatial feature map $\mathbf{F}_{\text{recon}}$ is passed through a decoder to generate the reconstructed image $\hat{\mathbf{I}}_q \in \mathbb{R}^{3\times H'\times W'}$. The decoder consists of:

- Multiple upsampling layers to progressively increase spatial resolution.
- Convolutional layers [1] with Instance Normalization [7] and ReLU activation to refine features.
- A projection layer that maps the features to the RGB space, followed by a sigmoid activation.

3.3. Effect of the Reconstruction Process

The reconstruction process ensures that keypoints are distributed across semantically meaningful regions of the object. By aggregating local features and incorporating global context, the reconstructed image retains both local structural details and global structural consistency. This optimization guarantees that the keypoints effectively capture the complete structure of the object.

4. More Implementation Details

Here we provide additional implementation details of our method. We use the transformer-based CroCoNetv2 [8] backbone, similar to DVMNet [10], as the feature extractor to obtain image features. The feature interaction module contains $L = 3$ attention blocks, whereas the structure-aware correspondence estimation module has $N = 4$ attention blocks. The model dimension C is 768, with each attention block utilizing 8 heads.

For the loss function, the total loss $\mathcal{L}_{\text{total}}$ consists of four components: the 3D keypoint loss $\mathcal{L}_{\text{pts}}$, the reconstruction loss $\mathcal{L}_{\text{rec}}$, the rotation loss $\mathcal{L}_{\text{rot}}$, and the mask loss $\mathcal{L}_{\text{mask}}$, as described in the main text. The corresponding weights are set to $\lambda_1 = 2$, $\lambda_2 = 1$, $\lambda_3 = 2$, and $\lambda_4 = 10$.

We train the network for 400 epochs, and the batch size is set to 80 on the CO3D dataset [6]. All experiments are

conducted on 4 RTX 3090 GPUs and an Intel Xeon Gold 6248R @ 4.000 GHz CPU. Our code is implemented using PyTorch 1.13.0 and CUDA 11.6.

5. More Visualizations

Figure 2 provides additional visualizations. The first column shows the original input images, followed by five heatmaps that depict the responses of image-specific keypoint detectors, while the last column presents the reconstructed images.

In the heatmap columns, the responses from image-specific keypoint detectors for each query highlight prominent structural elements of the objects. Strong activations are observed at these key points, indicating that our model effectively captures semantically and structurally significant regions, contributing to a robust understanding of the object.

The final column shows the reconstructed images, which demonstrate that our keypoint-based representation effectively captures essential structural features of the objects. These reconstructions retain key visual elements, validating the ability of our method to utilize structural information for accurate relative pose estimation.

References

[1] Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 2

[2] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13142–13153, 2023. 1

[3] Stefan Hinterstoisser, Vincent Lepetit, Slobodan Ilic, Stefan Holzer, Gary Bradski, Kurt Konolige, and Nassir Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Computer Vision–ACCV 2012: 11th Asian Conference on Computer Vision, Daejeon, Korea, November 5-9, 2012, Revised Selected Papers, Part I 11*, pages 548–562. Springer, 2013. 1

[4] Tomas Jakab, Ankush Gupta, Hakan Bilen, and Andrea Vedaldi. Unsupervised learning of object landmarks through conditional image generation. *Advances in neural information processing systems*, 31, 2018. 1

[5] Tejas D Kulkarni, Ankush Gupta, Catalin Ionescu, Sebastian Borgeaud, Malcolm Reynolds, Andrew Zisserman, and Volodymyr Mnih. Unsupervised learning of object keypoints for perception and control. *Advances in neural information processing systems*, 32, 2019. 1

[6] Jeremy Reizenstein, Roman Shapovalov, Philipp Henzler, Luca Sbordone, Patrick Labatut, and David Novotny. Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10901–10911, 2021. 1, 2

[7] D Ulyanov. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016. 2

[8] Philippe Weinzaepfel, Thomas Lucas, Vincent Leroy, Yohann Cabon, Vaibhav Arora, Romain Brégier, Gabriela Csurka, Leonid Antsfeld, Boris Chidlovskii, and Jérôme Revaud. Croco v2: Improved cross-view completion pre-training for stereo matching and optical flow. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17969–17980, 2023. 2

[9] Chen Zhao, Tong Zhang, and Mathieu Salzmann. 3d-aware hypothesis & verification for generalizable relative object pose estimation. *arXiv preprint arXiv:2310.03534*, 2023. 1

[10] Chen Zhao, Tong Zhang, Zheng Dang, and Mathieu Salzmann. Dvmnet: Computing relative pose for unseen objects beyond hypotheses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20485–20495, 2024. 1, 2