3D-AWARE HYPOTHESIS & VERIFICATION FOR GENERALIZABLE RELATIVE OBJECT POSE ESTIMATION SUPPLEMENTARY MATERIAL

Anonymous authors
Paper under double-blind review

1 ARCHITECTURE OF THE 3D REASONING MODULE

We show the architecture of the 3D reasoning module in Fig. 1. Each 3D reasoning block consists of a self-attention layer and a cross-attention layer, which excel at capturing intra-view and inter-view relationships, respectively. The input 2D feature map is flattened from $\mathbb{R}^{C\times H_f\times W_f}$ to $\mathbb{R}^{N\times C}$, where $N = H_f \times W_f$. A position embedding, denoted as PE, is added to the flattened feature map. Fig. 1(b) illustrates the attention layer. The context refers to the input feature map itself in the self-attention layer and it represents the feature map of another view in the cross-attention layer. We use the standard multi-head attention (Vaswani et al., 2017) and layer normalization (Ba et al., 2016) in our attention layers.

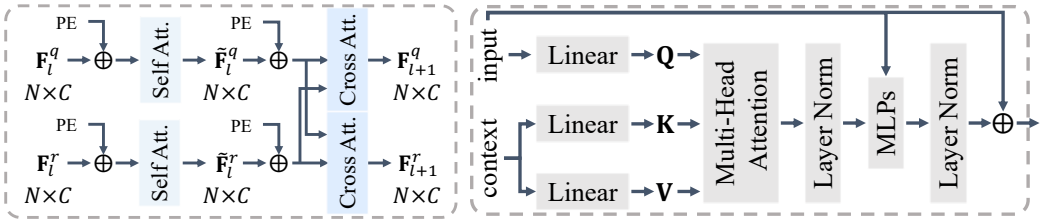
2 DATA CONFIGURATION

The synthetic images are generated by rendering objects of Objaverse from randomly sampled viewpoints (Liu et al., 2023). We attach these images to random backgrounds which are sampled from COCO (Lin et al., 2014). We randomly sample 128 objects from Objaverse and use 5 objects from LINEMOD sampled by Liu et al. (2022) as testing data, reserving the remaining objects for training. This design guarantees that all objects are previously unseen during the testing phase. We train the network on both synthetic and real data, alleviating the problem of domain gap.

Recall that we assume we have access to only one reference image and the objective is to estimate the relative object pose between the reference and the query. Therefore, the selection of the reference image is a crucial aspect of our benchmark. As multi-view images are available in Objaverse and LINEMOD datasets, one could randomly sample a reference given a query. However, such a strategy may yield an inappropriate reference. As shown in Fig. 2, the object depicted in the reference image barely overlaps with the one in the query, which makes the relative object pose estimation too challenging. Therefore, we filter out the inappropriate references from the datasets during training and testing, which makes our evaluation more reasonable.

Specifically, we convert the object rotation matrices $\mathbf{R}^r$ and $\mathbf{R}^q$ to Euler angles $(\alpha_r, \beta_r, \gamma_r)$ and $(\alpha_q, \beta_q, \gamma_q)$, which indicate azimuth, elevation, and in-plane rotation, respectively. Note that only azimuth and elevation lead to viewpoint changes, which thus determine the co-visible regions between the reference and query. Consequently, we set the in-plane rotation to 0 and convert the Euler angle back to the rotation matrix, i.e., $\tilde{\mathbf{R}} = h(\alpha, \beta, 0)$. We then measure the difference of the new rotation matrices $\tilde{\mathbf{R}}^r$ and $\tilde{\mathbf{R}}^q$ by computing the geodesic distance. We exclude the image pair with a distance larger than a predefined threshold ($90°$ by default in our experiments). As illustrated in Fig. 4 in our main paper, the retained image pairs display acceptable variations in object pose. Moreover, we utilize the synthetic images on Objaverse generated by Liu et al. (2023). Each 3D object model is rendered from 10 randomly sampled viewpoints, which yields synthetic images without in-plane rotations. To introduce in-plane rotations, we rotate the reference and query images using randomly sampled 2D in-plane rotations during training and testing.

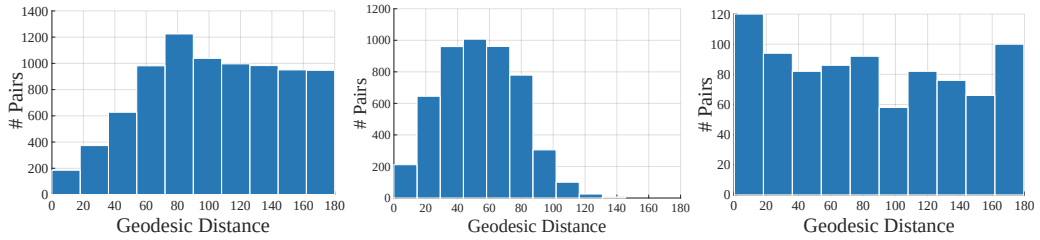
Fig. 3 shows the histograms of object pose variations between the reference and query images. We measure the variations based on the geodesic distance between the two object rotation matrices $\mathbf{R}^r$ and $\mathbf{R}^q$. The histograms show that the image pairs we used in our experiments exhibit a diverse range of object pose variations, which makes our evaluation results convincing.



(a) 3D reasoning block (b) Attention layer

Figure 1: **Architecture of the 3D reasoning module.**

Figure 2: **Examples of inappropriate references.**



(a) **Objaverse** (b) **LINEMOD** (c) **CO3D**

Figure 3: **Histograms of the object pose variation between the reference and query**. We measure the object pose variation as the geodesic distance between the two object rotation matrices $\mathbf{R}^r$ and $\mathbf{R}^q$. The histogram depicts the number of image pairs falling within different distance intervals.

3 Qualitative Results of 6D Object Pose Estimation

We extend our method to 6D pose estimation for unseen objects by utilizing an off-the-shelf generalizable object detector (Liu et al., 2022). More concretely, instead of using dense-view reference images, we feed the one reference we have in our benchmark to the pretrained detection network, which predicts the object bounding box in the query image. We use the parameters of the object bounding box to compute 3D object translation, following the implementation in (Liu et al., 2022). Subsequently, we crop the object from the query and employ our approach to predict the relative 3D object rotation. The object rotation in the query is derived as $\mathbf{R}^q = \Delta\mathbf{R}\mathbf{R}^r$. Fig. 4 shows some qualitative results of 6D pose estimation for the unseen objects on LINEMOD. We draw the 3D object bounding boxes in blue and green, using the predicted 6D object pose and the ground truth, respectively. The promising results demonstrate the potential of our approach in terms of generalizable 6D object pose estimation.

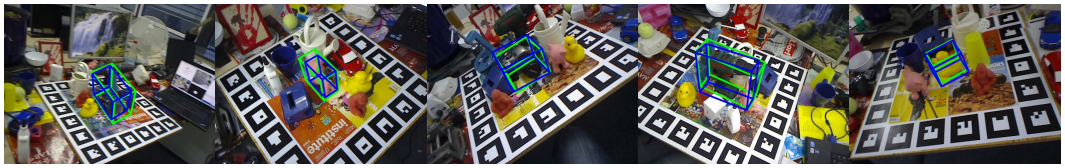


Figure 4: **Qualitative results of 6D pose estimation for unseen objects on LINEMOD.** The blue and green 3D object bounding boxes are drawn using the predicted 6D object pose and the ground truth, respectively.

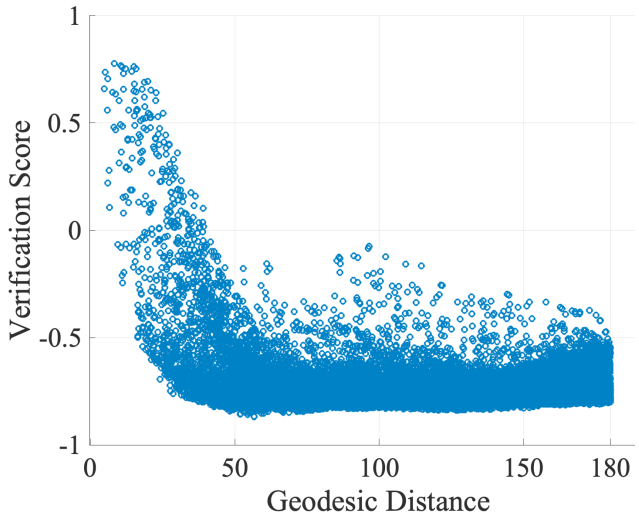


Figure 5: **Verification scores of all sampled pose hypotheses.** The x-axis and y-axis represent the geodesic distance between the pose samplings and the ground-truth relative object pose, and the verification scores, respectively.

4 More Detail about the Ablation Studies

As we introduced in the main paper, we performed an ablation study, evaluating the robustness against the noise added to the 2D object bounding boxes. We simulate the bounding boxes in real-world applications by performing jittering to the ground truth with different levels of noise. We denote the object center and the size of the bounding box as c and s. We then randomly sample the perturbed parameters from the intervals $(c - 0.5 * n * s, c + 0.5 * n * s)$ and $(\frac{s}{1+n}, s * (1 + n))$, respectively, where n indicates the noise. We varied n from 0.05 to 0.3 in our experiments. Please refer to Fig. 5 (b) in our main paper for the experimental results.

5 Efficiency

It is worth noting that during testing, our method utilizes 50,000 pose samples, while RelPose++ uses 500,000. Despite processing fewer samples, our method achieves better accuracy in relative object pose estimation. To further evaluate the efficiency, we measure the computation cost in multiply-accumulate operations (MACs) and show the results in Table 1. All evaluated methods process the pose samples in parallel. "RelPose++-5000 and "Ours-5000" refer to RelPose++ and our method with 5,000 samples, respectively. The results clearly show that our method achieves a better trade-off between efficiency and accuracy in relative object pose estimation. Additionally, our method with only 5,000 samples still delivers more accurate results than RelPose++ with 500,000 samples.

Method	RelPose++	**Ours**	RelPose++-5000	**Ours-5000**
MACs	94.6	54.7	11.3	16.3
Angular Error	38.5	28.5	50.7	35.3

Table 1: **Efficiency.** Relpose++ uses 500,000 pose samples by default, while we sample 50,000 poses for our method in our experiments. RelPose++-5000 and Ours-5000 denote RelPose++ and our method with 5,000 pose samples, respectively. The multiply-accumulate operations (MACs) are used to measure the computation consumption.

REFERENCES

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Proceedings of the European Conference on Computer Vision*, pp. 740–755. Springer, 2014.

Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. *arXiv preprint arXiv:2303.11328*, 2023.

Yuan Liu, Yilin Wen, Sida Peng, Cheng Lin, Xiaoxiao Long, Taku Komura, and Wenping Wang. Gen6d: Generalizable model-free 6-dof object pose estimation from rgb images. *Proceedings of the European Conference on Computer Vision*, 2022.

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.