Joint Learning of Pose Regression and Denoising Diffusion with Score Scaling Sampling for Category-level 6D Pose Estimation

Supplementary Material

In this supplementary document, we first provide implementation details of our method (Section S.1), followed by extensive experimental analyses that complement the main paper (Section S.2). Lastly, we show additional qualitative results that demonstrate our method's effectiveness across various object categories and challenging scenarios. (Section S.3).

S.1. Implementation Details

Architecture. Based on the architecture of GenPose [12], our network consists of PointNet++ [8], which extracts a 1024-dimensional global feature from the input point cloud. The time step t and noisy pose $p(t)$ are embedded through MLPs to produce 128-dimensional and 256-dimensional feature vectors, respectively. These three features are concatenated to a 1408-dimensional vector and fed into the denoising diffusion head to predict a 9D score vector (6D rotation representation and 3D translation). The regression head consists of MLP of size 1024×512×512×9, directly predicting the 6D representation and 3D translation vectors.

6D Rotation Representation. For rotation representation, we employ the continuous 6D representation following [12, 16], where g_{GS} maps SO(3) to 6D representation by retaining the first two columns of the rotation matrix:

$$g_{GS}\left(\begin{bmatrix} a_1 & a_2 & a_3 \end{bmatrix}\right) = \begin{bmatrix} a_1 & a_2 \end{bmatrix} \tag{1}$$

f_{GS} maps 6D representation back to SO(3) through Gram-Schmidt-like orthogonalization [16]:

$$f_{GS}\left(\begin{bmatrix} a_1 & a_2 \end{bmatrix}\right) = \begin{bmatrix} b_1 & b_2 & b_3 \end{bmatrix} \tag{2}$$

$$b_i = \begin{bmatrix} \begin{cases} N(a_1) & \text{if } i = 1 \\ N\big(a_2 - (b_1 \cdot a_2) b_1\big) & \text{if } i = 2 \\ b_1 \times b_2 & \text{if } i = 3 \end{cases} \end{bmatrix}^\top \tag{3}$$

Training Details. In the pre-training phase, we train only the PointNet++ encoder and regression head, where the regression head outputs 9D vectors (6D rotation and 3D translation). The network is optimized by first mapping the predicted 6D rotation to SO(3) to compute the geodesic loss and combining it with L2 loss for translation. We adopt the geodesic distance for rotation loss, as it provides a clearer learning goal on SO(3) compared to L2 loss [2]. Also, as shown in Figure S1, our experiments demonstrate superior performance with geodesic loss (blue) compared to L2 loss (red). For objects that are fully symmetric around the y-axis (bottle, bowl, and can) in the NOCS dataset [10], we compute the rotation loss only with the y-axis to account for their symmetry properties. In the joint learning phase, we initialize the encoder and regression head with the pre-trained weights and simultaneously train both regression and diffusion heads. The regression head maintains its pre-training loss function, while the diffusion head is optimized with the score-matching objective. The network is then updated using the sum of these two losses.

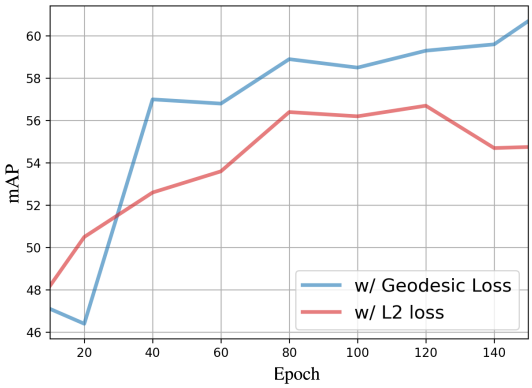


Figure S1. Comparison of pre-training performance on REAL275 dataset using different rotation loss functions for the regression head.

S.2. More Results and Analyses

S.2.1. Inference Efficiency with DDIM Sampler

Diffusion-based methods inherently suffer from computational burden due to multiple sampling steps required during inference. To address this limitation and enable explicit control over the number of sampling steps, we replace the original adaptive-step PF-ODE solver with a deterministic DDIM sampler [18], as mentioned in the main paper (Figure 4(d)). We perform sampling using the following DDIM update equation:

$$\begin{aligned} \mathbf{x}_{t-1} = \mathbf{x}_t &+ \sigma_t^2 \left(w_t \cdot \mathbf{s}_\theta(\mathbf{x}_t, t)\right) \\ &+ \sqrt{\sigma_{t-1}^2 - (\eta\sigma_{t-1})^2} \cdot \left(-\mathbf{s}_\theta(\mathbf{x}_t, t) \cdot \sigma_t\right) \\ &+ \eta\sigma_{t-1}\boldsymbol{\epsilon}_t \end{aligned} \tag{4}$$

where $\mathbf{x}_t$ and $\mathbf{x}_{t-1}$ represent a generated pose samples at time steps t and $t-1$ respectively, w_t is our proposed score

Method	Num steps	Speed(FPS)↑	5°2cm ↑	10°2cm ↑	Params(M)↓
Ours$^{\text{DDIM}}$	500	2.70	15.9	23.6	2.2
Ours$^{\text{DDIM+G}}$	500	2.70	52.7	72.7	2.2
Ours$^{\text{DDIM+G}}$	200	6.81	52.7	72.8	2.2
Ours$^{\text{DDIM+G}}$	100	14.1	52.6	72.7	2.2
Ours$^{\text{DDIM+G}}$	50	25.4	52.4	72.6	2.2
Ours$^{\text{DDIM+G}}$	10	42.9	50.8	71.3	2.2
HS-Pose [19]	1	50	46.5	68.6	6.1
Query6DoF [20]	1	34.9	49.0	68.7	19.4

Table S1. Comparison of forward-only methods and ours with different sampling steps on REAL275 dataset for single frame pose estimation. For Ours, $^{\text{DDIM}}$ indicates DDIM Sampler without score scaling, and $^{\text{DDIM+G}}$ indicates DDIM Sampler with score scaling guidance.

scaling weight, σ_t represents the noise level at time step t, $\mathbf{s}_\theta(\mathbf{x}_t, t)$ is the learned score function, η controls the stochasticity of sampling (set to 0 for deterministic sampling), and $\boldsymbol{\epsilon}_t \sim \mathcal{N}(0, \mathbf{I})$ is Gaussian noise.

As shown in Table S1, our method with score scaling guidance (Ours$^{\text{DDIM+G}}$) demonstrates remarkable robustness across different sampling steps for single pose estimation. Even with only 10 sampling steps, we achieve 42.9 FPS while maintaining competitive accuracy. Notably, the performance remains nearly constant from 500 steps down to 50 steps, highlighting the effectiveness of our score scaling approach. In contrast, performance drops dramatically without scoring scaling guidance (Ours$^{\text{DDIM}}$: 52.7 $\rightarrow$ 15.9, 72.7 $\rightarrow$ 23.6), further underscoring the critical role of our proposed guidance method.

Compared to recent forward-only baselines (HS-Pose [19] and Query6DoF [20]), our approach offers a favorable speed-accuracy trade-off while using significantly fewer parameters, thus making it well-suited for real-time applications.

S.2.2. Analysis of Pre-training strategies

Pre-training encoders has been widely adopted in various vision tasks to leverage learned representations, with notable success in models like Latent Diffusion Models (LDMs) [9] where pre-trained image encoders significantly reduce computational costs while maintaining generation quality. However, our investigation reveals that this established practice does not directly translate to category-level 6D pose estimation with point cloud inputs. While previous works like [1, 4, 6, 14, 15] have utilized various point cloud encoders (e.g., 3DGC [5], PointNet++ [8]) trained in an end-to-end fashion, we systematically evaluated different pre-training strategies within the GenPose framework to potentially accelerate convergence and enhance performance.

As shown in Figure 4(a) in the main paper, we compared various pre-training strategies on NOCS dataset: classification of object categories (orange), point cloud reconstruction (green), and direct 6D pose regression (purple). Surprisingly, both classification and reconstruction pre-training show marginal or even negative impact on convergence compared to training from scratch (While Figure 4(a) shows results with PointNet++ encoder, similar patterns were observed when using Transformer-based encoder [7]). In contrast, pre-training with direct 6D pose regression demonstrates notably faster convergence and better performance. We hypothesize that this phenomenon stems from the higher complexity of 6D pose estimation compared to classification or reconstruction tasks. Unlike classification which extracts category-discriminative features or reconstruction which preserves the local neighborhood structures [11], 6D pose estimation demands the encoder to learn both fine-grained geometric features and their global spatial relationships in SE(3) space. When pre-trained on other tasks, the encoder learns features that may be suboptimal or even counterproductive for pose estimation. This observation led to our final design choice of joint learning with pose regression, which effectively combines the benefits of pre-trained pose-aware features with diffusion-based distribution modeling.

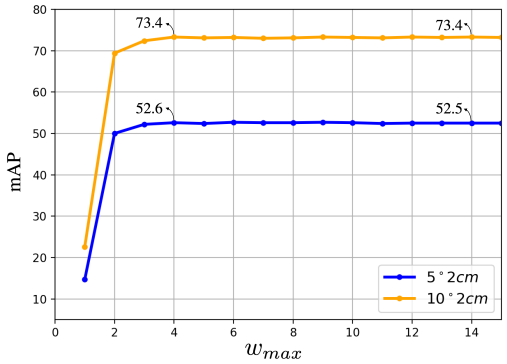


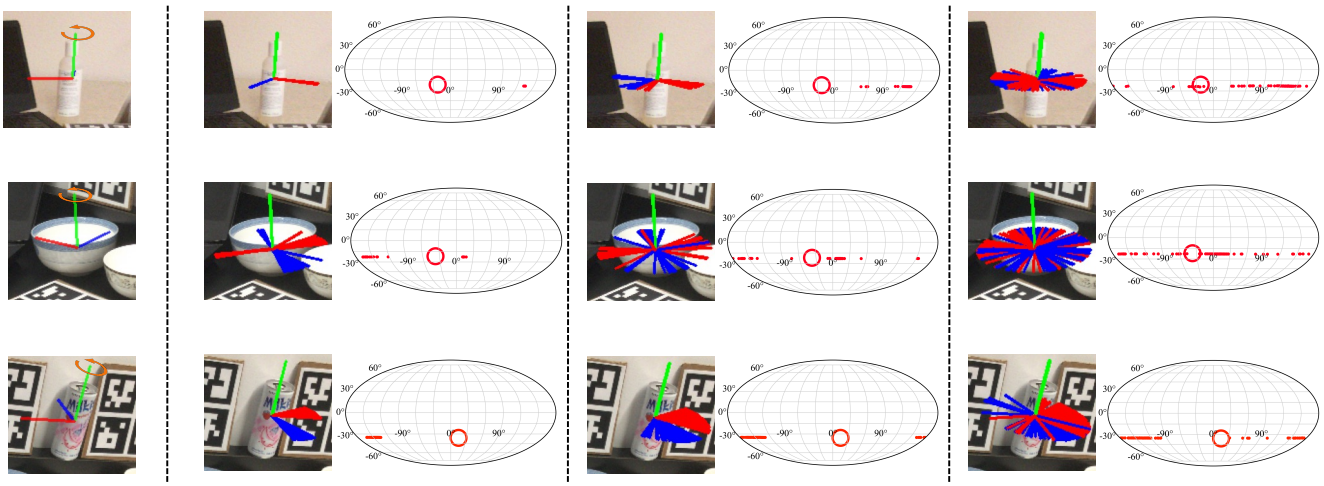
Figure S2. Performance on different weight parameters. All results are based on single pose sampling.

S.2.3. Ablation studies on Scaling Guidance

Comparison of Scaling Schedulers. To analyze the impact of guidance scheduling, we examine three different score scaling strategies:

$$w_t = \begin{cases} w_{max} + (w_{min} - w_{max})t & \text{(linear)} \\ w_{min} + (w_{max} - w_{min})\exp(-5t) & \text{(exponential)} \\ w_{max} & \text{(constant)} \end{cases} \tag{5}$$

Here, we set $w_{min} = 1.0$ and $w_{max} = 4.0$. Table S2 shows similar quantitative performance across different schedulers on REAL275 dataset, though Figure S3 reveals their distinct behaviors when handling symmetric objects. Using 50 identical random noise inputs, the constant scheduler (a) leads to mode collapse with strong convergence to specific modes. The linear scheduler (b), which gradually increases guidance weight, shows improved but still limited



Ground-truth (a) Constant scaling (b) Linear scaling (c) Exponential scaling

Figure S3. Comparison of rotation distributions for 50 sampled poses across different scaling schedules.

capability in capturing the full symmetric distribution. In contrast, the exponential scheduler (c), which begins with weak guidance and gradually increases over time, best preserves the symmetric distribution while maintaining pose fidelity. This observation highlights that weak guidance in early time steps is crucial for exploring the symmetric distribution space, while stronger guidance near $t \to 0$ helps improve pose quality.

Guidance Scheduler	5°2cm	5°5cm	10°2cm	10°5cm
Constant	52.7	61.2	73.2	84.5
Linear	52.7	61.1	73.2	84.4
Exponential	52.6	61.0	73.4	84.5

Table S2. Results on REAL275 dataset according to score scaling weight scheduler.

Choice of the weight parameter for Guidance. We analyze the impact of the maximum weight parameter w_{max} on model performance while fixing $w_{min} = 1$ and using exponential scheduling. As shown in Figure S2, pose estimation accuracy improves as w_{max} increases from 1 to 4 on the REAL275 dataset ($w_{max} = 1$ refers to w/o Guidance). However, the performance plateaus beyond $w_{max} = 4$, with minimal or no improvements for larger values. We set $w_{max} = 4$ as our default value based on these empirical results.

S.2.4. Generalization ability

Following GenPose [12], we evaluate our model on unseen categories in REAL275, focusing on symmetric objects. As presented in Table S3, our method exhibits generalization capabilities comparable to those of GenPose, whereas other baseline [17] suffers significant performance degradation. This indicates that diffusion-based models can generalize effectively to unseen object categories, particularly when they share geometric structures with the training set.

Category	Method	5°2cm	5°5cm	10°2cm	10°5cm
bowl	SAR-Net	58.1/36.4	66.0/47.3	83.7/59.4	93.6/81.5
	GenPose	85.4/64.5	92.6/72.5	93.1/87.2	100.0/98.6
	Ours	87.4/65.7	93.4/72.3	93.9/88.4	100.0/99.1
bottle	SAR-Net	43.5/11.7	54.0/23.0	61.3/33.6	79.8/68.0
	GenPose	52.6/39.0	60.9/53.2	81.4/73.6	92.7/94.6
	Ours	54.8/42.1	65.4/57.0	81.3/74.4	93.4/93.1

Table S3. Cross-category results on REAL275. Left and right of '/' denote seen and unseen category performance, respectively.

S.3. Additional Qualitative Results

Figure S4 provides a qualitative comparison between our method and the baseline (GenPose), illustrating their distinct sampling processes. The baseline's approach (shown in pink dashed box) requires multiple steps: (1) sampling K(=50) pose candidates, (2) filtering out low-likelihood poses using an additional EnergyNet, and (3) computing the final pose through mean pooling the remaining 30 poses. In contrast, our method (shown in green dashed box) employs score scaling guidance to generate high-quality poses with just a single pose sampling, eliminating the need for multiple pose candidates and additional filtering networks.

Figure S5 presents additional qualitative comparisons between our method and GenPose on the HouseCat6D. GenPose (columns 1-2) first samples 50 pose candidates and computes the final output through filtering and mean pooling. While our method (columns 3-4) also samples 50 poses for comparison with the baseline, we randomly select

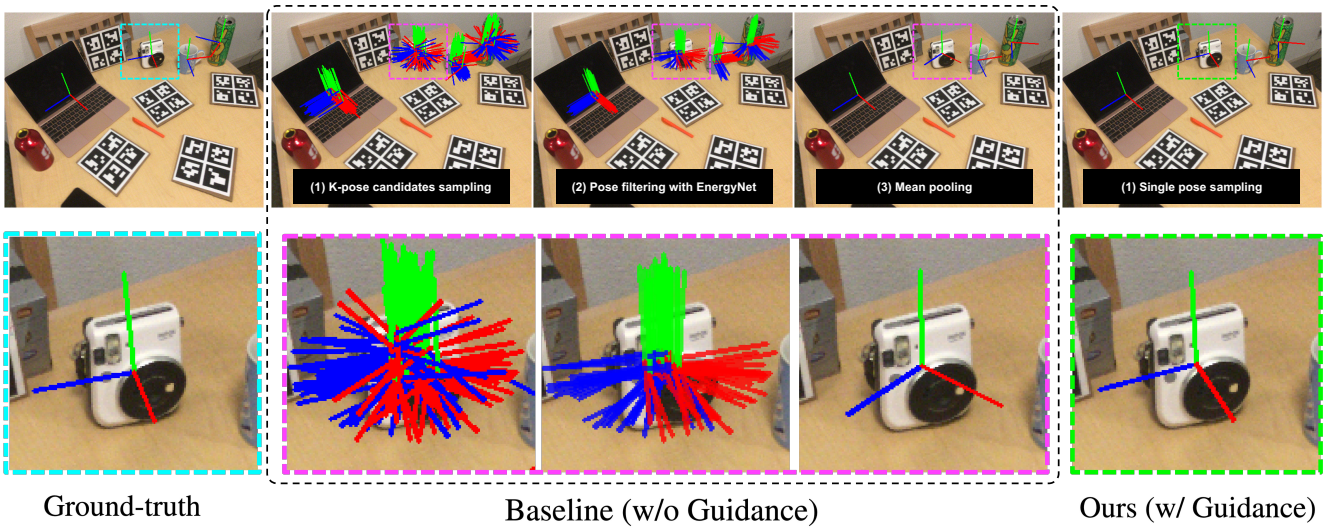


Figure S4. Qualitative comparison of Ours (column 5) and Baseline (column 2∼4)

a single pose to visualize the final output, demonstrating the effectiveness of our score scaling guidance. For asymmetric objects (Box, Teapot, Shoe), while the baseline generates outlier samples that deviate from the ground truth pose (column 1), our method consistently produces pose samples that closely align with the ground truth (column 3). This improvement can be attributed to our guidance method, which effectively steers the sampling process toward high-density regions of the pose distribution. The results for symmetric objects, particularly the Glass example, further highlight the advantages of our method. The baseline shows scattered pose samples around the symmetric axis, while our method with score scaling guidance accurately captures the object's symmetry. Specifically, our method precisely identifies the y-axis as the axis of symmetry (column 3), maintaining appropriate pose diversity while ensuring high-quality predictions.

Figure S6 further demonstrates qualitative results on the ROPE dataset, highlighting our method's capability to handle objects with discrete symmetries. In the case of 'Boxed beverage', which inherently has two symmetric ground truth poses (front and back), our method (column 3) successfully captures both valid pose modes. This demonstrates that our score scaling guidance effectively preserves the multi-modal nature of the pose distribution when dealing with objects that have multiple ground truth poses due to symmetry.

References

[1] 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 CVPR, 2022. 2

[2] Ge Gao, Mikko Lauri, Yulong Wang, Xiaolin Hu, Jianwei Zhang, and Simone Frintrop. 6d object pose regression via supervised learning on point clouds. In ICRA, 2020. 1

[3] HyunJun Jung, Shun-Cheng Wu, Patrick Ruhkamp, Guangyao Zhai, Hannah Schieber, Giulia Rizzoli, Pengyuan Wang, Hongcheng Zhao, Lorenzo Garattoni, Sven Meier, et al. Housecat6d-a large-scale multi-modal category level 6d object perception dataset with household objects in realistic scenarios. In CVPR, 2024.

[4] Xiao Lin, Wenfei Yang, Yuan Gao, and Tianzhu Zhang. Instance-adaptive and geometric-aware keypoint learning for category-level 6d object pose estimation. In CVPR, 2024. 2

[5] Zhi-Hao Lin, Sheng-Yu Huang, and Yu-Chiang Frank Wang. Convolution in the cloud: Learning deformable kernels in 3d graph convolution networks for point cloud analysis. In CVPR, 2020. 2

[6] Jianhui Liu, Yukang Chen, Xiaoqing Ye, and Xiaojuan Qi. Prior-free category-level pose estimation with implicit space transformation. In ICCV, 2023. 2

[7] Yatian Pang, Wenxiao Wang, Francis EH Tay, Wei Liu, Yonghong Tian, and Li Yuan. Masked autoencoders for point cloud self-supervised learning. In ECCV, 2022. 2

[8] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In NeurIPS, 2017. 1, 2

[9] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Bj¨orn Ommer. High-resolution image synthesis with latent diffusion models. In CVPR, 2022. 2

[10] 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 CVPR, 2019. 1

[11] Yaoqing Yang, Chen Feng, Yiru Shen, and Dong Tian. Foldingnet: Point cloud auto-encoder via deep grid deformation. In CVPR, 2018. 2

[12] Jiyao Zhang, Mingdong Wu, and Hao Dong. Genpose: Generative category-level object pose estimation via diffusion models. In NeurIPS, 2023. 1, 3

[13] Jiyao Zhang, Weiyao Huang, Bo Peng, Mingdong Wu, Fei Hu, Zijian Chen, Bo Zhao, and Hao Dong. Omni6dpose: A benchmark and model for universal 6d object pose estimation and tracking. In ECCV, 2025.

[14] Ruida Zhang, Yan Di, Zhiqiang Lou, Fabian Manhardt, Federico Tombari, and Xiangyang Ji. Rbp-pose: Residual bounding box projection for category-level pose estimation. In ECCV, 2022. 2

[15] 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. In IROS, 2022. 2

[16] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In CVPR, 2019. 1

[17] 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 CVPR, 2022. 3

[18] Jiaming Song, Chenlin Meng and Stefano Ermon. Denoising diffusion implicit models. In ICLR, 2021. 1

[19] 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 CVPR 2023. 2

[20] 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 ICCV 2023. 2

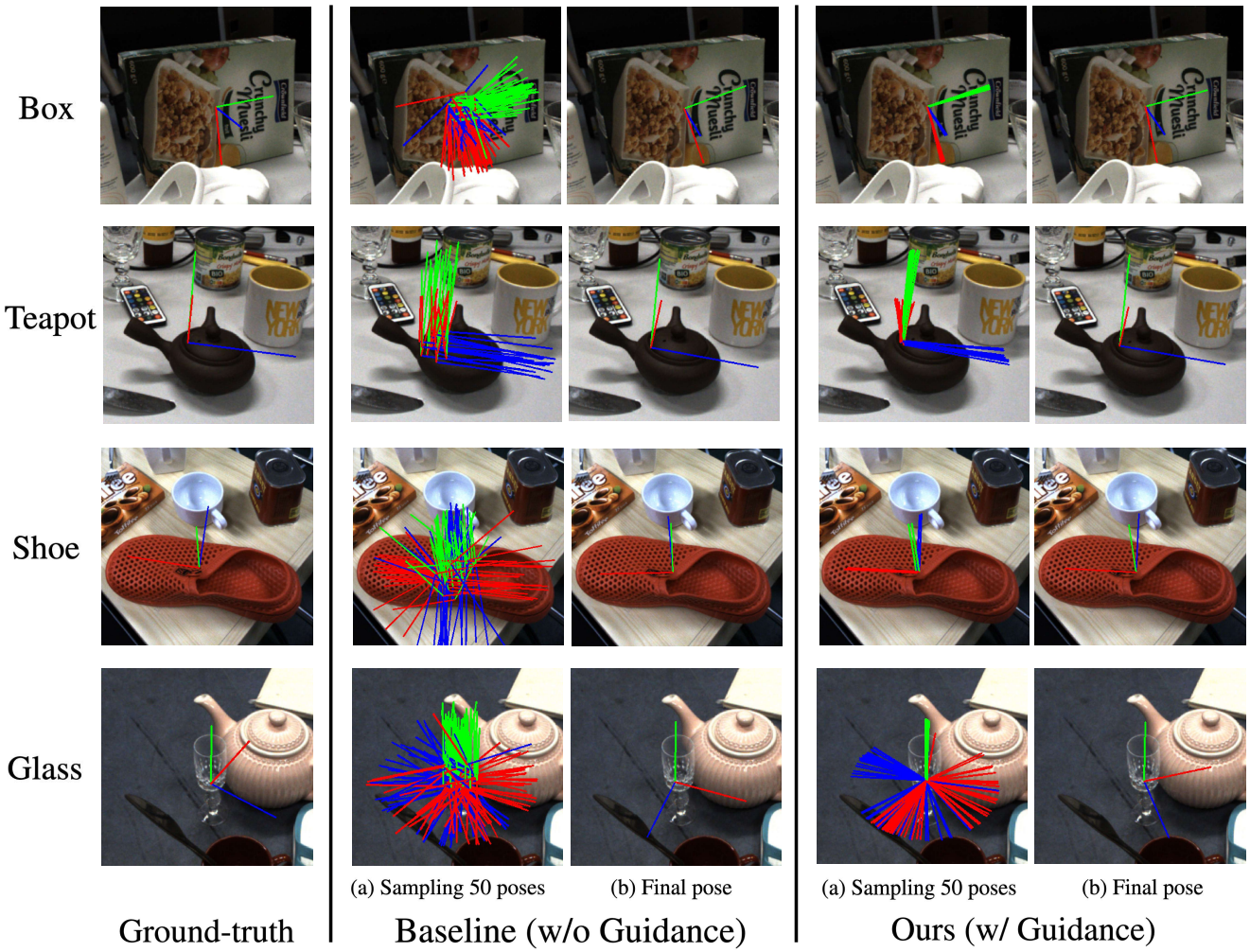


Figure S5. Qualitative comparison between Ours (column 3∼4) and Baseline (column 1∼2) on HouseCat6D.

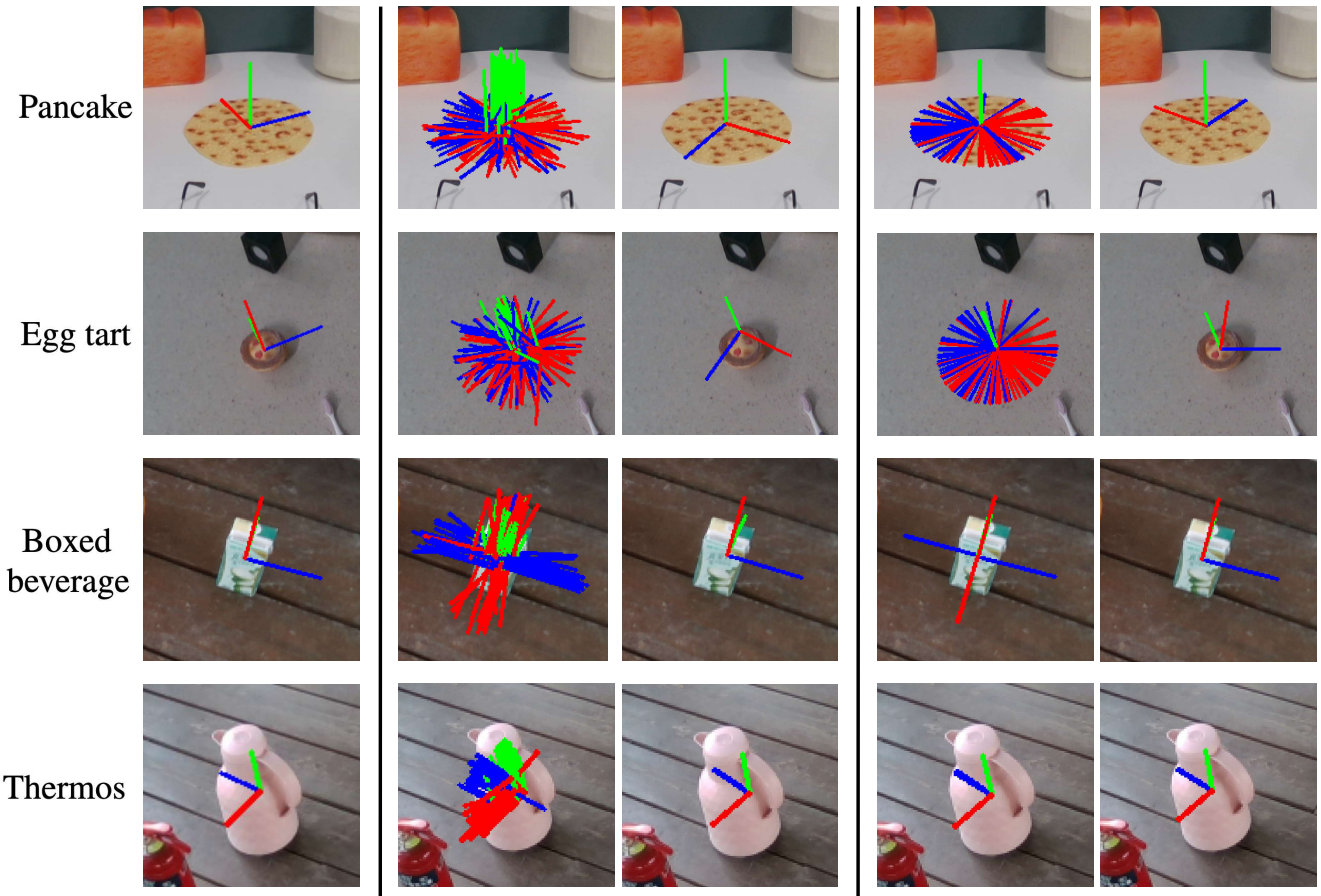


Figure S6. Qualitative comparison between Ours (column 3∼4) and Baseline (column 1∼2) on ROPE.