Two-stage Monocular 6D Pose Estimation for Small Cubic Objects

Xinmiao Du[1] , Zhuoyu Jiang[1] and Xihong Wu[1*]

[1] School of Intelligence Science and Technology
Key Laboratory of Machine Perception (MoE)
Peking University, Beijing 100871, China
2001111389@stu.pku.edu.cn, 2401112131@stu.pku.edu.cn,
wxh@cis.pku.edu.cn

Abstract. Monocular 6D object pose estimation is an important perception problem for robotic manipulation. For high-precision operations such as block grasping, alignment, and placement, the perception module must provide sufficiently accurate and stable pose estimates rather than coarse object localization alone. This requirement is particularly challenging for small cubic objects due to weak texture, limited visual cues, strong rotational symmetry, and sensitivity to Region of Interest (ROI) quality. In this paper, we study monocular 6D pose estimation of small cubic objects from a single RGB image and propose a two-stage manipulation-oriented framework. In the first stage, a detection-guided ROI-based regression model is used to estimate object pose under symmetry-aware supervision. In the second stage, we further explore a MuJoCo-based re-rendering strategy to construct consistency-enhanced training samples for refinement, aiming to improve adaptation to manipulation-relevant visual conditions. Experiments on a unified MuJoCo-generated dataset show that, on the main single-block benchmark, the proposed method achieves the strongest overall balance in ADD-S, translation accuracy, rotation stability, and task-oriented usability metrics. Preliminary results in multi-block scenes further suggest favorable generalization to more complex visual conditions.

Keywords: Monocular 6D Pose Estimation, High-Precision Robotic Manipulation, Physics-based Re-rendering Consistency[1]

1 Introduction

Monocular 6D pose estimation is a core perception problem in robotic manipulation: high-precision tasks such as block grasping and placement demand centimeter or sub-centimeter accuracy beyond what coarse localisation can provide. Small cubic objects compound the difficulty by combining weak texture, strong rotational symmetry, and a small image footprint that magnifies detection and cropping errors. Existing monocular method[21, 12, 6, 11, 20], render-and-compare

1 Corresponding author: wxh@cis.pku.edu.cn

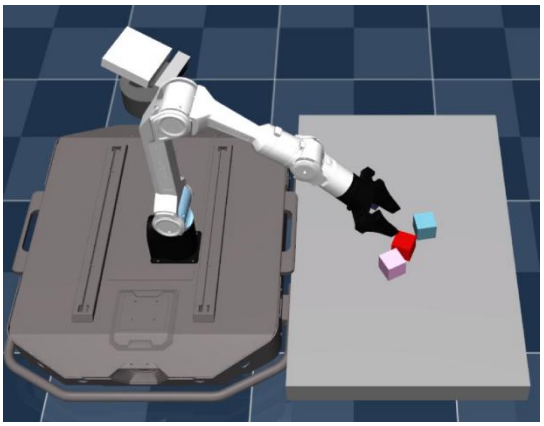

Fig. 1. MuJoCo-based tabletop scenario. The red cube is the primary pose-estimation target; the other cubes provide clutter, proximity, and occlusion for the preliminary multi-block evaluation.

refinement[8], and manipulation-oriented systems[18] mostly target textured or geometrically distinctive objects, while pseudo-labelling and consistency-based training under limited supervision[19, 1, 14] trade performance against supervision cost and deployment robustness.

We propose a two-stage monocular framework for this setting (Fig. 1 illustrates the tabletop scenario). Stage 1 performs detection-guided ROI-based pose regression; Stage 2 refines the prediction through a MuJoCo-based re-rendering consistency mechanism[17] that uses the simulation environment, camera intrinsics, and CAD model as training-time physical priors. We additionally evaluate the framework in preliminary multi-block scenes.

Our contributions are threefold. (i) We frame the task as manipulation-tolerance-driven and introduce grasp-feasibility (Grasp-F) and placement-feasibility (Place-F) metrics that encode centimeter and sub-centimeter tolerances complementary to ADD-S. (ii) We combine detection-guided ROI regression and 24-element symmetry-aware supervision with a Stage-2 re-rendering consistency mechanism; unlike Self6D[19] and TexPose[1], the consistency branch uses a non-differentiable MuJoCo renderer with a geometric-validity filter and a perturbed-pose target, controlling error accumulation without the engineering cost of differentiable rendering. (iii) On a unified MuJoCo benchmark the method yields consistent gains under both GT-ROI and predicted-ROI settings, with preliminary multi-block evidence of generalization.

2 Related Work

2.1 Monocular 6D Pose Estimation

Existing monocular 6D pose methods fall into three groups. Direct regression predicts rotation and translation from image or region features[21, 6], with later variants disentangling pose components[9]. Keypoint-based methods recover pose via PnP from predicted 2D landmarks[12], but landmark detection becomes unstable for weakly textured and highly symmetric objects. Dense-correspondence methods, from Pix2Pose[11] to DPOD[22], EPOS[5] and ZebraPose[15], establish denser image-to-surface mappings; geometry-guided regressors such as GDR-Net[20] reach strong results on textured or geometrically distinctive objects, yet their advantages diminish on small, textureless, strongly symmetric targets.

Symmetry and weak texture, the two factors dominating our setting, have been studied separately. Pitteri et al.[13] formalize how symmetries break standard rotation supervision and propose symmetry-aware losses; ES6D[10] pursues this direction with grouped primitives but relies on RGB-D input; the T-LESS benchmark[3] has driven progress on texture-less industrial parts. Most of these works, however, evaluate on objects substantially larger than ours, where the joint effect of symmetry, weak texture, and ROI sensitivity on small cubes remains less systematically addressed---this gap motivates the manipulation-tolerance-driven evaluation we adopt.

2.2 Semi-supervised and Consistency-based Learning

Pseudo-labeling and consistency-based training have been explored to mitigate label scarcity in pose estimation. The former expands supervision through teacher--student self-training, while the latter encourages stable predictions under perturbations or rendering-based supervision[14], as exploited by pose-specific methods such as Self6D[19] and TexPose[1]; both lines, however, trade off performance against supervision cost and deployment robustness.

Render-and-compare refinement[8, 7] iteratively aligns rendered hypotheses with the input image but typically assumes a differentiable or tightly coupled rendering loop, and domain randomisation[16] together with synthetic-to-real benchmarks such as BOP[4] provides a complementary simulation-as-prior route. Our Stage 2 inherits this simulation-as-prior view but uses a non-differentiable renderer with a perturbed-pose target rather than image alignment, which we find sufficient for small symmetric cubes and keeps the training pipeline simple.

3 Method

3.1 Problem Formulation

Given a single RGB image $I \in \mathbb{R}^{H\times W\times 3}$, the camera intrinsics K, and the CAD model M of the target cube, the goal is to estimate $P = (R, t)$ with $R \in SO(3)$ and $t \in \mathbb{R}^3$ in the camera frame. The main single-block task predicts the absolute pose of the target; all reported errors reflect perception only, with no robot control or execution loop involved. An overview of the framework is shown in Fig.2.

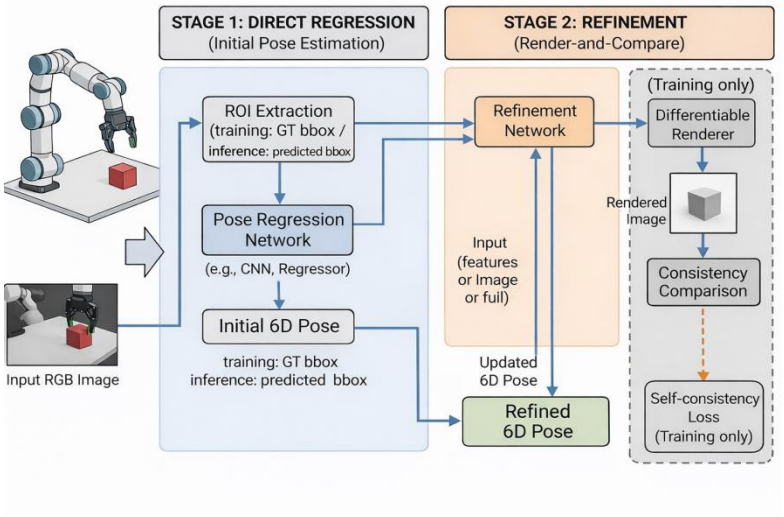

Fig. 2. MOverview of the proposed two-stage framework.**Blue panel (Stage 1):** a detection-guided ROI extractor crops the target region (GT bbox during training, predicted bbox during inference); a pose-regression network outputs an initial 6D pose. **Orange panel (Stage 2):** a refinement network takes the initial pose and ROI features as input and produces a refined pose. **Gray dashed panel (training only):** a non-differentiable MuJoCo renderer generates an additional image from a perturbed pose, and a self-consistency loss aligns the network's prediction on the rendered image with the perturbed pose. The dashed branch is removed at inference, so the deployed pipeline relies only on monocular RGB input.

3.2 Symmetry-aware Pose Supervision

In Stage 1, a lightweight detector predicts the target bounding box and a visibility score; the cropped ROI is then fed into a pose-regression branch that outputs translation and rotation. Detection and pose share the backbone and are trained jointly, which reduces background interference for small targets compared with full-image regression and yields more consistent features than a separately cascaded detect-then-pose pipeline.

3.3 Symmetry-aware Pose Supervision

Since the target object is a cube, its rotation exhibits strong symmetry, and standard rotation regression would incorrectly treat multiple equivalent poses as different targets. To address this issue, we adopt symmetry-aware rotation supervision based on the 24-element rotational symmetry group $G = \{S_1, \dots, S_{24}\}$ of the cube, where each $S \in G$ acts on the ground-truth rotation as R^*S, so that all 24 visually equivalent ground-truth poses are treated as equally valid supervision targets.

Let $\hat{t}$,t^*denote the predicted and ground-truth translations, and $\hat{R}$, $R^*$the predicted and ground-truth rotations. The translation loss adopts the Smooth L_1 form:

$$L_{trans} = SmoothL1(\hat{t} - t^*) \tag{1}$$

The rotation loss is defined as the minimum geodesic distance over all symmetry-equivalent ground-truth rotations:

$$L_{rot}^{sym} = \min_{S \in G} arccos\left(\frac{tr(\hat{R}^\top R^* S) - 1}{2}\right) \tag{2}$$

In practice, the argument of $arccos$ is clamped to $[-1 + \epsilon, 1 - \epsilon]$ for numerical stability. The bounding box loss L_{bbox} uses Smooth L_1 on the four box coordinates, and the visibility loss L_{vis} uses binary cross-entropy on the predicted visibility score. The overall supervised objective in Stage 1 is then

$$L \quad = L_{trans} + \lambda_r L_{rot}^{sym} + \lambda_b L_{bbox} + \lambda_v L_{vis} \tag{3}$$

3.4 Stage 2: MuJoCo-based Re-rendering Consistency Training

Stage 2 inherits the Stage-1 weights and continues to optimise L jointly with a consistency term on samples re-rendered from the model's own current predictions. Given I_A, the Stage-1 model predicts $\hat{P}_A$ together with a visibility score $\hat{v}_A$. A sample enters the consistency branch only when $\hat{R}_A$ remains a proper rotation after re-orthogonalisation, the depth of $\hat{t}_A$ lies within the tabletop range, the projected box has a moderate side length, and $\hat{v}_A$ exceeds a confidence threshold; this filter prevents grossly incorrect initial predictions from being reinforced.

A small perturbation $(\Delta R, \Delta t)$ is sampled and applied to the retained $\hat{P}_A$ to obtain $\tilde{P} = (\tilde{R}, \tilde{t})$. MuJoCo then re-renders I_B from $\tilde{P}$ using the known intrinsics and CAD model, after which the network performs a second forward pass on I_B to predict $\hat{P}_B = (\hat{R}_B, \hat{t}_B)$. The consistency loss takes the same form as the supervised loss, but uses $\tilde{P}$ as its target:

$$L_{unsup} = SmoothL1(\hat{t}_B - \tilde{t}) + \lambda_r \min_{S \in G} arccos\left(\frac{tr(\hat{R}_B^\top \tilde{R} S) - 1}{2}\right) \tag{4}$$

Because $\tilde{P}$ is by construction the exact pose used to render I_B, this loss carries no label noise of the type that plagues pseudo-labels on I_A. Systematic bias in $\hat{P}_A$can still skew the distribution of $\tilde{P}$; the geometric-validity filter together with the retained L jointly anchor training against this drift.

The overall training objective is

$$L = L \quad + \lambda_u L_{unsup} \tag{5}$$

where λ_u follows a sigmoid ramp-up from 0 to λ_u^{max} over the first 30% of Stage 2 training, ensuring the consistency branch contributes meaningfully only after the pose head has stabilised. Unlike Self6D[19] and TexPose[1], our renderer is non-differentiable---gradients flow only through the network's prediction on I_B, not through rendering---which keeps the simulation as a training-time physical prior without the engineering cost of differentiable rendering. At inference the consistency branch is removed; the deployed network operates on monocular RGB through the Stage 1 detection and pose-regression branches alone, and the same architecture is reused without modification in the multi-block evaluation as a generalisation test rather than a new component.

4 Experiments

4.1 Dataset and Experimental Protocol

The main experiments are conducted on a MuJoCo-generated dataset for single-block pose estimation. The dataset contains 10,000 samples; 9,000 are used for training and 1,000 form a fixed test set, split by a fixed random seed and shared across all methods. Each scene contains a single cubic target on a tabletop. The camera viewpoint is sampled on a hemisphere around the table, with elevation, azimuth, and distance covering typical tabletop manipulation viewpoints; the target block is randomly placed within a bounded tabletop region with yaw uniformly sampled in $[0°, 360°]$. Domain randomisation is applied to lighting intensity and direction, tabletop and background textures, and small camera-roll perturbations. Pose annotations (R,t) are recorded directly from the simulator state, so labels are noise-free.

A supervision ratio of $r\%$ samples $r\%$ of the 9,000 training images uniformly without replacement as labelled data; the remainder is unlabelled and visible only to methods that exploit it (Pseudo-label and ours). Sampled indices are fixed across baselines, and reported numbers are averaged over three random seeds unless noted. All results use 224×224 input under symmetry-aware evaluation, with GT boxes as a controlled diagnostic (Table 1) and predicted and perturbed boxes for deployable usage (Table 3; perturbed boxes apply controlled offsets to GT). Multi-block scenes serve as a generalisation test rather than a primary benchmark.

4.2 Evaluation Metrics and Baselines

Since the target is a symmetric cube, we use ADD-S as the primary metric, with ADD reported only as a supplementary geometric metric, alongside translation error, rotation error, Recall@1cm, and Recall@5deg. To reflect manipulation-oriented requirements, we further define two task-oriented feasibility metrics as threshold-based proxies derived from pose error rather than closed-loop execution success: a prediction is grasp-

feasible (Grasp-F) when translation error is below 1cm and rotation error below $10°$, and placement-feasible (Place-F) under the tighter thresholds 0.5cm and $5°$.

We compare against **Stage1-only** (Stage 1 loss only), **ROI direct regression**, **Pseudo-label** (confidence-thresholded self-training in the spirit of[14], adapted to pose), **Keypoint+PnP** (eight cube corners followed by EPnP, in the keypoint paradigm of PVNet[12] and PVN3D[2]), and a **GDR-Net-style** regressor[20], covering direct-regression, pseudo-label, keypoint, and geometry-based pipelines.

4.3 Implementation Details

All methods share a CNN backbone with an ROI-aligned pose head (two MLP branches for translation and rotation) and a lightweight detection head. We train with AdamW (lr 1×10^{-4}, cosine schedule, weight decay 10^{-4}), batch size 32, and input 224×224; Stage 1 runs for 60 epochs and Stage 2 fine-tunes for 30 more from the Stage 1 weights. The supervised weights are $\lambda_r = 1.0$, $\lambda_b = 1.0$, $\lambda_v = 0.5$, and λ_u follows a sigmoid ramp-up from 0 to $\lambda_u^{max} = 0.5$ over the first 30% of Stage 2 (stable for $\lambda_u^{max} \in$). Baselines share the backbone, schedule, and splits, with method-specific choices following each baseline's original recipe: *Pseudo-label* uses a confidence threshold of 0.7 on the visibility score with no rotation-specific filtering; Keypoint+PnP regresses the eight cube-corner 2D coordinates with a Smooth-L_1 loss and recovers pose via OpenCV's *solvePnP* (EPnP, RANSAC); *GDR-Net-style* reproduces the dense-correspondence head of [20] on the same backbone with a Patch-PnP head and surface-region attention, but no symmetry-aware re-supervision; *ROI-DR* shares Stage 1's ROI extractor while omitting joint detection training and the Stage-2 consistency term.

4.4 Main Results on the Single-block Benchmark

We organise the single-block evaluation around two complementary settings. Table 3 presents the deployment-relevant results under predicted and perturbed bounding boxes, which reflect actual usage when no oracle box is available; Table 1 then provides a controlled diagnostic under GT-ROI to isolate the pose-estimation behaviour from detector noise across supervision ratios. Reading the two together avoids conflating perception accuracy with detector quality, and the conclusions agree on both views.

Under the deployment setting (Table 3), our method leads on rotation accuracy ($6.83°$ vs. $14.61°$), Recall@5deg (0.582 vs. 0.163), and grasp-feasibility (0.256 vs. 0.136), with translation degrading by only 0.20cm relative to the GT-box upper bound---evidence that the method does not depend on perfect localisation. Under the GT-ROI diagnostic (Table 1), 40% is treated as the main supervision ratio. Our method achieves the strongest overall balance at 10% and 40%, and at 20% it retains a substantial rotation advantage (11.82° vs. 22.29°) and leads on all feasibility metrics, even though Pseudo-label is slightly better on ADD-S (0.0070 vs.0.0074) and translation (1.32 vs. 1.40cm). This trade-off is interpretable: confidence-filtered pseudo-labels tighten translation locally, while rotation on textureless cubes remains under-determined without explicit symmetry-aware re-supervision. The gap that $\mathcal{P}$ closes is visible across budgets:

rotation error under our method drops from 11.82° at 20% to 6.84° at 40% while translation changes only mildly, and even ROI direct regression at 40%, despite reaching 9.09° rotation error, lags on feasibility---confirming that ROI cropping alone does not meet manipulation tolerances without Stage 2 refinement. Fig. 3 traces these trends across label budgets.

Table 1 . GT-ROI diagnostic table: single-block results across supervision ratios with oracle bounding boxes, isolating pose accuracy from detector noise. Deployment-relevant results under predicted boxes are reported in Table 3 . ADD-S is in meters, Trans in cm, Rot in degrees; R@1 denotes recall under translation error <1cm and R@5 under symmetry-aware rotation error <5°; G-F requires translation <1cm and rotation <10°, and P-F requires <0.5cm and <5°.

Method	Ratio	ADD-S	Trans	Rot	R@1	R@5	G-F	P-F
Stage1	10%	0.0136	2.55	19.82	0.039	0.160	0.007	0.001
Pseudo	10%	0.0098	1.86	19.84	0.148	0.163	0.046	0.004
ROI-DR	10%	0.0130	2.47	22.44	0.037	0.108	0.007	0.000
Ours	10%	**0.0083**	**1.58**	**18.65**	**0.236**	**0.187**	**0.088**	**0.014**
Stage1	20%	0.0085	1.61	22.72	0.219	0.064	0.048	0.003
Pseudo	20%	**0.0070**	**1.32**	22.29	0.324	0.093	0.074	0.009
ROI-DR	20%	0.0107	2.02	22.39	0.126	0.102	0.029	0.002
Ours	20%	0.0074	1.40	**11.82**	**0.343**	**0.340**	**0.215**	**0.028**
Stage1	40%	0.0071	1.35	17.42	0.300	0.082	0.061	0.001
Pseudo	40%	0.0063	1.19	13.23	0.406	0.230	0.192	0.012
ROI-DR	40%	0.0079	1.52	9.09	0.235	0.431	0.181	0.020
Ours	40%	**0.0061**	**1.15**	**6.84**	**0.461**	**0.573**	**0.394**	**0.067**

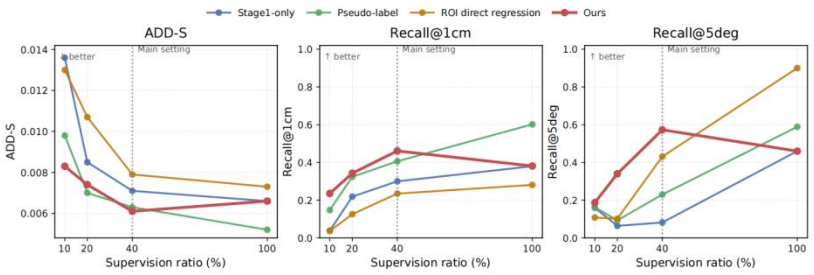

Fig. 3. Effect of supervision ratio on single-block pose estimation, GT-ROI protocol. Three metrics are shown side by side: ADD-S (lower is better), Recall@1cm (translation tolerance, higher is better) and Recall@5deg (symmetry-aware rotation tolerance, higher is better). Each curve corresponds to one method; the vertical dashed line marks the main 40% supervision setting reported in Table 1 . The 100% point is shown only as an upper-bound reference and is not the focus of comparison. The proposed method dominates Recall@5deg across all budgets and matches Pseudo-label on ADD-S/Recall@1cm only at low budgets, where labelled rotation supervision is most under-determined.

4.5 Ablation on the Two-stage Design

Table 2 . Ablation results under different Stage-2 training settings.

Setting	ADD-S	Trans	Rot	R@5	G-F	P-F
Stage1-only	0.0071	1.35	17.42	0.082	0.061	0.001
Ours w/o RC	**0.0060**	**1.13**	7.52	0.524	0.401	0.076
Ours short-stage2	0.0256	4.13	12.30	0.304	0.003	0.000
Ours full-stage2	0.0061	1.15	**6.84**	0.573	0.394	0.067

Table 2 isolates the contribution of Stage 2. The collapse of Ours short-stage2 (ADD-S 0.0256, worse than Stage1-only) reflects an unfinished ramp-up: when Stage 2 is truncated before λ_u saturates, the consistency loss dominates the gradient before the pose head has stabilised, pulling the model off the supervised optimum---a transient that disappears when the schedule is allowed to complete (Ours full-stage2). The remaining gain from render consistency concentrates on rotation, reducing error from 7.52° to 6.84° and improving Recall@5deg from 0.524 to 0.573, while ADD-S and translation already approach saturation under the two-stage structure alone---consistent with rotation being the under-constrained component on textureless cubes, for which $\hat{P}$ supplies supervision that labelled data alone cannot.

4.6 Robustness under Different ROI Protocols

Table 3 . Comparison under different ROI protocols. Trans denotes translation error (cm) and Rot rotation error (deg); G-F is the grasp-feasibility rate. GT, Pred., and Pert.refer to ground-truth, predicted, and perturbed bounding boxes, respectively.

ROI	Method	ADD-S	Trans	Rot	R@1	R@5	G-F
GT	Pseudo	0.0063	1.19	13.23	0.406	0.230	0.192
GT	Ours	**0.0061**	**1.15**	**6.84**	**0.461**	**0.573**	**0.394**
Pred.	Pseudo	**0.0068**	**1.30**	14.61	**0.350**	0.163	0.136
Pred.	Ours	0.0070	1.35	**6.83**	0.313	**0.582**	**0.256**
Pert.	Pseudo	0.0068	1.29	13.26	0.353	0.233	0.157
Pert.	Ours	**0.0067**	**1.27**	**6.85**	**0.370**	**0.579**	**0.301**

Across the three ROI settings (Table 3), our method's rotation error remains at $6.83° - 6.85°$, whereas Pseudo-label fluctuates between 13.23° and 14.61° and degrades by $1.4°$ from GT to predicted boxes---indicating that the symmetry-aware L_{rot}^{sym} and $\hat{P}$-derived targets decouple rotation supervision from box quality, while pseudo-label rotation supervision implicitly relies on tight cropping. Translation, by contrast, inflates

more under predicted (1.15 → 1.35cm) than perturbed (1.27cm) boxes: detector outputs occasionally introduce systematic offsets that random perturbations do not, identifying detector quality---rather than a method-side limitation---as the dominant remaining source of translation residual.

4.7 Preliminary Generalization to Multi-block Scenes

In multi-block scenes (Table 4; only the red cube's pose is scored), our method consistently outperforms baselines across three seeds. The large GDR-Net-style errors (45.87cm, 36.09°) exceed the physical scene scale: when visually identical cubes coexist, geometry-guided correspondences cannot disambiguate instance identity, and predictions collapse onto a wrong cube---putting error at the order of inter-cube distances rather than within-cube residuals. Methods anchored on a global target, such as ours, avoid this failure by design.

Table 3 . Comparison under different ROI protocols. Trans denotes translation error (cm) and Rot rotation error (deg); G-F is the grasp-feasibility rate. GT, Pred., and Pert.refer to ground-truth, predicted, and perturbed bounding boxes, respectively.

Method	Trans	Rot
Stage1-only	3.61	4.63
ROI direct regression	1.40	4.64
Keypoint+PnP	10.16	20.80
GDR-Net-style	45.87	36.09
Ours (seed1)	**0.37**	3.10
Ours (seed2)	0.39	**2.70**
Ours (seed3)	0.40	2.83

5 Conclusion

We presented a two-stage monocular 6D pose framework for small symmetric cubes that targets manipulation-relevant precision through detection-guided ROI regression, 24-symmetry-aware supervision, and a Stage-2 MuJoCo re-rendering consistency mechanism whose perturbed-pose target sidesteps the label noise typical of pseudo-labels. The framework attains strong ADD-S, translation, rotation, and feasibility scores on the single-block benchmark, with preliminary multi-block evidence of generalisation. Residual sensitivity to ROI quality, Stage~2 training-budget allocation, and the synthetic-only training distribution remain the main limitations; real-world validation, cross-dataset evaluation on standard texture-less benchmarks, more stable sample selection, and multi-object extensions are natural next steps.

References

1. Chen, H., Manhardt, F., Navab, N., Busam, B.: TexPose: Neural Texture Learning for Self-Supervised 6D Object Pose Estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6662–6671 (2023)
2. He, Y., Sun, W., Huang, H., Liu, J., Fan, H., Sun, J.: PVN3D: A Deep Point-Wise 3D Keypoints Voting Network for 6DoF Pose Estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 11629–11638 (2020)
3. Hodaň, T., Haluza, P., Obdržálek, Š., Matas, J., Lourakis, M., Zabulis, X.: T-LESS: An RGB-D Dataset for 6D Pose Estimation of Texture-less Objects. In: IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 880–888 (2017)
4. Hodaň, T., Sundermeyer, M., Drost, B., Labbé, Y., Brachmann, E., Michel, F., Rother, C., Matas, J.: BOP Challenge 2020 on 6D Object Localization. In: Bartoli, A., Fusiello, A. (eds.) Computer Vision – ECCV 2020 Workshops. LNCS, vol. 12536, pp. 577–594. Springer, Cham (2020)
5. Hodaň, T., Barath, D., Matas, J.: EPOS: Estimating 6D Pose of Objects with Symmetries. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 11703–11712 (2020)
6. Hu, Y., Fua, P., Wang, W., Salzmann, M.: Single-Stage 6D Object Pose Estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2930–2939 (2020)
7. Labbé, Y., Carpentier, J., Aubry, M., Sivic, J.: CosyPose: Consistent Multi-view Multi-object 6D Pose Estimation. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) Computer Vision – ECCV 2020. LNCS, vol. 12362, pp. 574–591. Springer, Cham (2020)
8. Li, Y., Wang, G., Ji, X., Xiang, Y., Fox, D.: DeepIM: Deep Iterative Matching for 6D Pose Estimation. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) Computer Vision – ECCV 2018. LNCS, vol. 11210, pp. 695–711. Springer, Cham (2018)
9. Li, Z., Wang, G., Ji, X.: CDPN: Coordinates-Based Disentangled Pose Network for Real-Time RGB-Based 6-DoF Object Pose Estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 7678–7687 (2019)
10. Mo, N., Gan, W., Yokoya, N., Chen, S.: ES6D: A Computation Efficient and Symmetry-Aware 6D Pose Regression Framework. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6708–6717 (2022)
11. Park, K., Patten, T., Vincze, M.: Pix2Pose: Pixel-wise Coordinate Regression of Objects for 6D Pose Estimation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 7668–7677 (2019)
12. Peng, S., Liu, Y., Huang, Q., Zhou, X., Bao, H.: PVNet: Pixel-Wise Voting Network for 6DoF Pose Estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4561–4570 (2019)
13. Pitteri, G., Ramamonjisoa, M., Ilic, S., Lepetit, V.: On Object Symmetries and 6D Pose Estimation from Images. In: Proceedings of the International Conference on 3D Vision (3DV), pp. 614–622 (2019)
14. Sohn, K., Berthelot, D., Carlini, N., Zhang, Z., Zhang, H., Raffel, C., Cubuk, E.-D., Kurakin, A., Li, C.-L.: FixMatch: Simplifying Semi-Supervised Learning with Consistency and Confidence. In: Advances in Neural Information Processing Systems (NeurIPS) (2020)
15. Su, Y., Saleh, M., Fetzer, T., Rambach, J., Navab, N., Busam, B., Tombari, F.: ZebraPose: Coarse to Fine Surface Encoding for 6DoF Object Pose Estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6738–6748 (2022)

16. Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., Abbeel, P.: Domain Randomization for Transferring Deep Neural Networks from Simulation to the Real World. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 23–30 (2017)
17. Todorov, E., Erez, T., Tassa, Y.: MuJoCo: A Physics Engine for Model-Based Control. In: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 5026–5033 (2012)
18. Tremblay, J., To, T., Sundaralingam, B., Xiang, Y., Fox, D., Birchfield, S.: Deep Object Pose Estimation for Semantic Robotic Grasping of Household Objects. In: Conference on Robot Learning (CoRL), pp. 306–316 (2018)
19. Wang, G., Manhardt, F., Shao, J., Ji, X., Navab, N., Tombari, F.: Self6D: Self-Supervised Monocular 6D Object Pose Estimation. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) Computer Vision – ECCV 2020. LNCS, vol. 12346, pp. 108–125. Springer, Cham (2020)
20. Wang, G., Manhardt, F., Tombari, F., Ji, X.: GDR-Net: Geometry-Guided Direct Regression Network for Monocular 6D Object Pose Estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 16611–16621 (2021)
21. Xiang, Y., Schmidt, T., Narayanan, V., Fox, D.: PoseCNN: A Convolutional Neural Network for 6D Object Pose Estimation in Cluttered Scenes. In: Robotics: Science and Systems (RSS) (2018)
22. Zakharov, S., Shugurov, I., Ilic, S.: DPOD: 6D Pose Object Detector and Refiner. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 1941–1950 (2019)