AXISPOSE++: REFERENCE-ANCHORED TRI-AXIS TRANSPORT FOR SINGLE-REFERENCE UNSEEN OBJECT POSE ESTIMATION

Anonymous authors
Paper under double-blind review

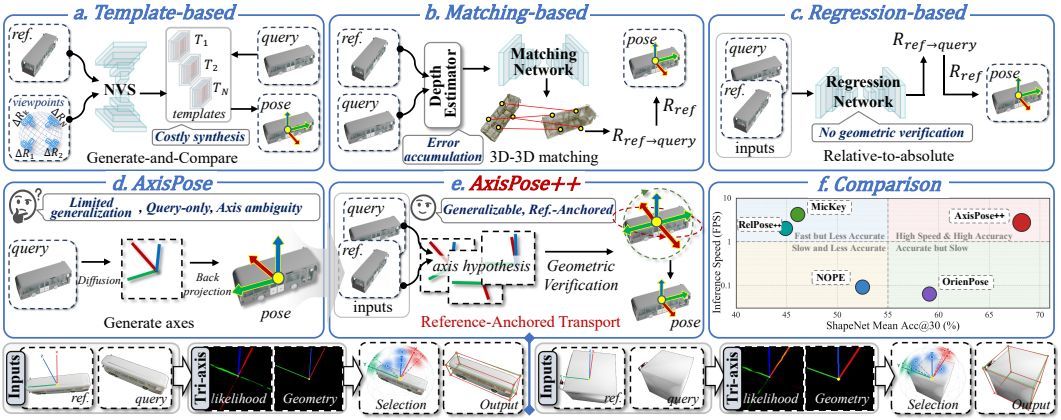

Figure 1: Existing methods primarily exploit the reference as visual evidence through costly novel-view synthesis (a), depth/correspondence-based matching (b), or direct regression without explicit geometric verification (c). AxisPose predicts query tri-axis projections but lacks reference anchoring and remains axis-ambiguous (d). AxisPose++ changes the role of the reference from visual evidence to a coordinate anchor: it transports the anchored tri-axis structure to the query and resolves multimodal axis hypotheses through geometric verification (e). Bottom visualizations show the predicted tri-axis likelihoods and global geometry, selected hypotheses, and final pose projections across different object appearances and shapes. The resulting formulation achieves a favorable accuracy–efficiency trade-off (f).

ABSTRACT

Estimating the unseen object's pose from minimal visual observations is essential for scalable robotic perception. While single-reference methods have shown promise, existing approaches typically rely on novel-view template synthesis, predicted geometry and local matching, or direct relative-pose regression. These pipelines remain sensitive to occlusion and weak texture and may accumulate errors through intermediate predictions. More fundamentally, the reference is primarily used as visual evidence rather than as an explicit coordinate anchor. We present AxisPose++, a reference-anchored paradigm for unseen object pose estimation from a single RGB reference. AxisPose++ formulates the task as coordinate transport: the posed reference anchors the object coordinate frame, whose compact tri-axis structure is directly transported to the target view. A geometry-grounded back-projection module then recovers the target rotation by enforcing axis orthogonality and camera-projection consistency. By avoiding costly intermediate synthesis and correspondence construction, AxisPose++ enables efficient geometry-grounded rotation recovery. Extensive experiments on synthetic and real-world benchmarks demonstrate strong generalization while being $43\times$ faster than OrienPose. Anonymous code is available at AxisPose++.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

Estimating the pose of previously unseen objects is fundamental to scalable robotic perception Liu et al. (2024); Zou et al. (2026b); Liu et al. (2026); Nguyen et al. (2024a). Conventional pose-estimation systems often rely on strong object-specific or sensing assumptions, including CAD models Wen et al. (2024); Nguyen et al. (2024b); Moon et al. (2025); Chen et al. (2026), depth observations Lee et al. (2025); Lin et al. (2024b); Wen et al. (2023); Hamza et al. (2026), or category-specific training Li et al. (2025); Lin et al. (2023); Liu et al. (2023). Although effective, such requirements hinder deployment to newly encountered objects whose geometry, depth, or category-specific supervision is unavailable. This motivates generalizable RGB-only pose estimation, which replaces object-specific priors with reference observations to generalize to unseen objects.

A major line of generalizable RGB-only methods relies on multiple posed reference views. Retrieval-based approaches, such as Gen6D Liu et al. (2022) and its successors Zhao et al. (2024); Pan et al. (2024), retrieve reference images with viewpoints similar to the query to initialize pose estimation and subsequently refine the result. Matching-based methods, including OnePose Sun et al. (2022) and its successors He et al. (2022), reconstruct object geometry from sparse reference images and establish 2D-3D correspondences with the query for pose recovery. BoxDreamer Yu et al. (2025) further represents object geometry using compact 3D bounding-box primitives for matching. Despite their progress, these methods still rely on multiple reference views and the quality of reconstruction and matching, making the single-reference setting particularly challenging.

Recent advances have begun to relax the reference requirement to a single RGB image. Generate-and-compare methods, such as NOPE Nguyen et al. (2024a) and OrienPose Liu et al. (2026), synthesize reference-conditioned novel-view templates and estimate pose by comparing them with the query image. Matching-based approaches, e.g., SingRef6D Wang et al. (2026) and MicKey Barroso-Laguna et al. (2024), predict geometric cues from RGB images and perform depth-aware feature matching. Other methods directly regress relative pose or pose distributions from the reference-query pair Du et al. (2022). These approaches demonstrate the feasibility of single-reference pose estimation, yet their performance remains constrained by synthesis artifacts, matching errors, and the limited out-of-distribution generalization of direct regressors.

The aforementioned approaches advance pose estimation beyond instance-specific training and CAD-model dependence and perform well when reliable visual or geometric evidence can be obtained from RGB observations. However, as shown in Figure 1 (a-c), existing single-reference methods still exhibit several practical bottlenecks: 1) **Appearance-template dependence.** Generate-and-compare methods Nguyen et al. (2024a); Liu et al. (2026) primarily use the reference as an appearance template. Their performance therefore depends on synthesis fidelity and reliable cross-view appearance cues, which can degrade under object symmetry, self-occlusion, or weak texture. 2) **Sensitivity to prediction errors.** Matching-based methods recover pose through predicted depth and local correspondences, whose errors can propagate to subsequent geometric solving under incomplete observations Qi et al. (2025); Wang et al. (2026). Direct-regression methods instead infer relative pose directly from the reference-query pair, without an explicit geometric verification step to detect or correct inconsistent predictions Du et al. (2022). 3) **High inference overhead.** Novel-view generation, depth prediction, and dense feature matching introduce considerable computational overhead (e.g., NOPE and OrienPose generate 342 templates for every single query image). These limitations motivate a reference-anchored representation that transports global coordinate structure to the target view and enables efficient geometry-consistent pose recovery.

To this end, we propose **AxisPose++**, a reference-anchored paradigm for single-reference unseen object pose estimation through tri-axis generation. As shown in Figure 1 (d), AxisPose Zou et al. (2026b) showed that an object's pose can be represented by the 2D projections of its three orthogonal canonical axes. Building on this representation, AxisPose++ treats the reference observation as a *coordinate anchor*, rather than an appearance template, and transfers its object-coordinate structure to the target view, as shown in Figure 1 (e).

Specifically, AxisPose++ consists of three stages, as shown in Figure 1 (bottom). First, *Reference-Coordinate Anchoring (RCA)* integrates the reference appearance, known pose, and tri-axis projection into a coordinate-aware condition that explicitly establishes the reference-defined object frame. Second, *Target-View Tri-Axis Transport (T^3)* propagates this coordinate structure to the target observation. A Multi-scale Coordinate-Control Injection (MCCI) mechanism injects the reference

condition into a lightweight U-Net at multiple feature scales, while a dual-head Tri-axis Likelihood Field (TLF) decoder predicts both dense tri-axis likelihood fields and complementary global axis geometry. This formulation avoids novel-view synthesis, depth prediction, and explicit cross-view feature matching. Finally, *Geometry-Grounded Pose Recovery (GGPR)* converts the predicted tri-axis representation into pose hypotheses. Inspired by the C2 algorithm Zou et al. (2026a), Tri-Axis Constrained Back-Projection (TACB) enforces axis orthogonality and camera-projection consistency, while Hypothesis-Aware Axis Selection (HAAS) resolves ambiguous axis responses and selects the most geometrically plausible pose. This yields an efficient 3D pose recovery process without correspondence-based PnP/RANSAC solving, as shown in Figure 1 (f).

Our contributions are as follows:

- We present **AxisPose++**, a reference-anchored paradigm for single-reference unseen object 3D pose estimation that reformulates reference-based pose inference as coordinate transport rather than appearance matching. By representing the reference-defined object frame using compact tri-axis projections, AxisPose++ achieves strong performance on synthetic and real-world benchmarks while being 43$\times$ faster than OrienPose.
- We introduce a **Target-View Tri-Axis Transport** framework that propagates reference-defined coordinate structure through multi-scale coordinate conditioning and jointly predicts dense tri-axis likelihood fields and global axis geometry. This avoids novel-view synthesis, depth estimation, and explicit cross-view feature correspondences.
- We develop a **Geometry-Grounded Pose Recovery** scheme that constructs 3D pose hypotheses through tri-axis constrained back-projection and resolves ambiguous axis predictions through geometry-based selection, enabling efficient and geometry-consistent inference without correspondence-based PnP/RANSAC pose solving.

2 Related Work

Generalizable RGB-Only Pose Estimation. Retrieval-based approaches select viewpoint-similar references for pose initialization and refinement Liu et al. (2022); Zhao et al. (2024), whereas matching-based approaches reconstruct sparse or semi-dense object geometry for 2D-3D correspondence estimation Sun et al. (2022); He et al. (2022). BoxDreamer Yu et al. (2025) instead uses reconstructed 3D bounding-box corners as compact correspondence primitives. These methods generalize to unseen objects but retain the cost of multi-view acquisition, reconstruction, or pose annotation. Recent methods reduce the input to a single RGB reference by directly regressing relative transformations Du et al. (2022), comparing pose-conditioned latent viewpoint embeddings Nguyen et al. (2024a) or synthesized novel views Liu et al. (2026), and recovering metric geometry through predicted depth Wang et al. (2026) or learned 3D correspondences Barroso-Laguna et al. (2024). AxisPose++ instead formulates single-reference pose estimation as reference-anchored coordinate transport, avoiding per-query view generation, depth prediction, and local feature matching.

Representations for Generalizable Pose Estimation. An effective intermediate representation is essential for generalizable object pose estimation Yu et al. (2025). Existing methods encode object identity, appearance, or geometry in different forms. Canonical-coordinate methods map observations to a normalized object space shared within a category Wang et al. (2019); Chen et al. (2020); Tian et al. (2020). Geometry-based methods represent objects using sparse or semi-dense point clouds Sun et al. (2022); He et al. (2022), multi-view feature volumes Liu et al. (2022), or neural implicit fields Wen et al. (2024). Other representations encode viewpoint-dependent appearance through latent embeddings Nguyen et al. (2024a) or synthesized views Liu et al. (2026), and lift monocular observations into metric geometry through depth maps Wang et al. (2026) or metric 3D keypoints Barroso-Laguna et al. (2024). More compact representations describe pose using global structural primitives, such as projected 3D bounding-box corners Rad & Lepetit (2017); Yu et al. (2025) or projected object tri-axes Zou et al. (2026b). AxisPose++ advances this direction by representing the instance-specific coordinate frame defined by a posed reference through its compact tri-axis projection, without category-level canonical priors or explicit object-shape reconstruction.

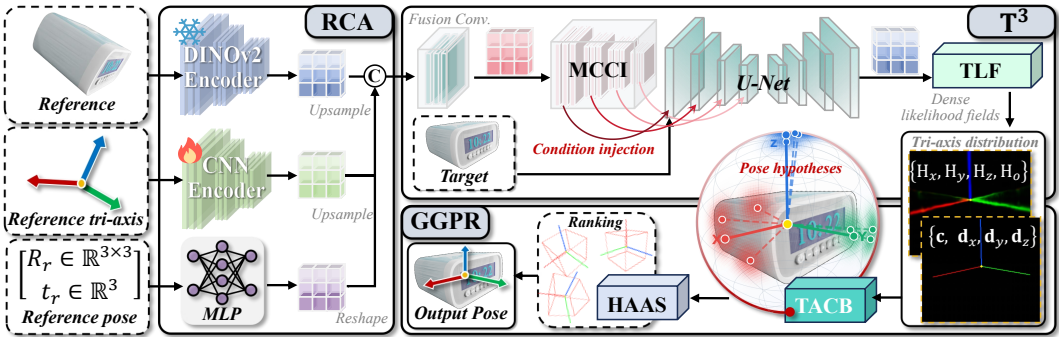

Figure 2: **Overview of AxisPose++.** Given a posed RGB reference, its projected tri-axis, and a target RGB image, AxisPose++ follows three stages. **Reference-Coordinate Anchoring (RCA)** integrates the reference appearance, tri-axis structure, and pose embedding into a coordinate-aware condition. **Target-View Tri-Axis Transport (T^3)** injects this condition into the target-view network through Multi-scale Coordinate-Control Injection (MCCI), while the Tri-axis Likelihood Field (TLF) decoder predicts dense axis likelihoods $\mathcal{H} = \{H_x, H_y, H_z, H_o\}$ and global axis geometry $\mathcal{G} = \{\mathbf{c}_\mathcal{G}, \mathbf{d}_x^\mathcal{G}, \mathbf{d}_y^\mathcal{G}, \mathbf{d}_z^\mathcal{G}\}$. Finally, **Geometry-Grounded Pose Recovery (GGPR)** uses Tri-Axis Constrained Back-Projection (TACB) to generate geometrically valid pose hypotheses and Hypothesis-Aware Axis Selection (HAAS) to resolve ambiguous axis predictions and select the final target pose.

3 Method

3.1 Problem Formulation and Overview

We consider single-reference rotation estimation for a previously unseen object, without access to CAD models, depth observations, or additional reference views. Given a posed RGB reference $(\mathbf{I}_r, \mathbf{K}_r, \mathbf{T}_r)$ and a target RGB image $(\mathbf{I}_t, \mathbf{K}_t)$, AxisPose++ estimates the target rotation $\mathbf{R}_t \in \mathrm{SO}(3)$ with respect to the object coordinate frame anchored by the reference. Here, $\mathbf{K}_r$ and $\mathbf{K}_t$ denote the camera intrinsics, and $\mathbf{T}_r = (\mathbf{R}_r, \mathbf{t}_r) \in \mathrm{SE}(3)$ is the known reference pose.

A single reference reveals only partial appearance and geometry of an unseen object, making direct cross-view matching inherently ambiguous under occlusion or weak texture. Rather than synthesizing additional views Liu et al. (2026); Nguyen et al. (2024a) or reconstructing object geometry Barroso-Laguna et al. (2024); Wang et al. (2026), AxisPose++ uses the known reference pose to establish an explicit object-coordinate anchor. Specifically, we define a canonical tri-axis $\mathcal{X}_{\text{axis}} = \{\mathbf{0}, \ell\mathbf{e}_x, \ell\mathbf{e}_y, \ell\mathbf{e}_z\}$, where ℓ is a fixed axis length, and project it onto the reference view as $\mathcal{A}_r = \Pi(\mathcal{X}_{\text{axis}}; \mathbf{K}_r, \mathbf{T}_r)$, where $\Pi(\cdot)$ denotes perspective projection. As shown in Figure 2, AxisPose++ follows three stages: Reference-Coordinate Anchoring (RCA), Target-View Tri-Axis Transport (T^3), and Geometry-Grounded Pose Recovery (GGPR), which are detailed below.

3.2 Reference-Coordinate Anchoring

Reference appearance alone does not specify how its visual evidence is related to the object coordinate frame. Reference Coordinate Anchoring (RCA) therefore converts the posed reference into an object-specific coordinate anchor by integrating its appearance, projected tri-axis, and known pose.

As shown in Figure 2, a frozen DINOv2 encoder Oquab et al. (2023) extracts semantically robust reference appearance features, while a lightweight convolutional encoder preserves the spatial structure of the aligned tri-axis map $\mathcal{A}_r$. An MLP embeds the flattened reference transformation $\mathrm{vec}([\mathbf{R}_r \mid \mathbf{t}_r])$ and broadcasts the resulting vector over the feature grid. After resizing the appearance and tri-axis features to a common resolution, RCA concatenates the three representations and applies convolutional fusion, yielding $\mathbf{C}_r = \mathcal{F}_{\text{RCA}}(\mathbf{I}_r, \mathcal{A}_r, \mathbf{T}_r)$. Through the pose-derived and image-aligned tri-axis, $\mathbf{C}_r$ explicitly ties reference appearance to the object coordinate frame and serves as the conditioning signal for target-view coordinate transport.

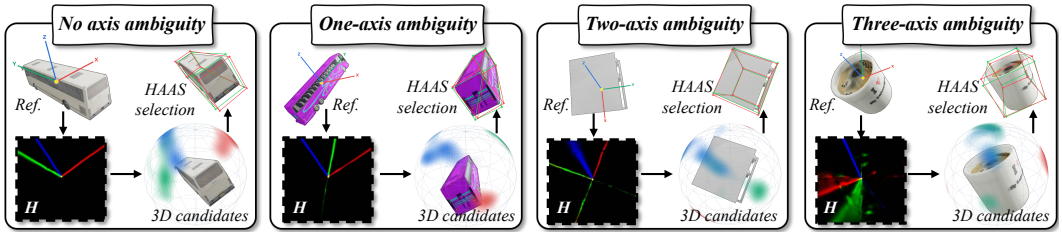

Figure 3: **Tri-axis ambiguity.** Weak texture or symmetry yields multimodal 2D evidence and multiple 3D rotation candidates. Red, green, and blue densities on the unit sphere show the candidate x-, y-, and z-axis directions, respectively. HAAS selects the geometrically consistent hypothesis.

3.3 Target-View Tri-Axis Transport

The second stage, Target-View Tri-Axis Transport (T^3), transports the reference-defined coordinate anchor to the target view. We use Multi-scale Coordinate-Control Injection (MCCI) to inject both local spatial cues and global structural context into the target features.

Multi-scale Coordinate-Control Injection. A lightweight U-Net Ronneberger et al. (2015); He et al. (2016) extracts multi-scale target features $\{\mathbf{X}_t^s\}_{s=1}^4$. In parallel, MCCI transforms $\mathbf{C}_r$ into a condition pyramid $\{\mathbf{C}_r^s\}_{s=1}^4$ matched to the target scales. At each encoder level, $\widetilde{\mathbf{X}}_t^s = \mathbf{X}_t^s + \mathcal{Z}_s(\mathbf{C}_r^s)$, where $\mathcal{Z}_s$ is a zero-initialized 1×1 convolution. This preserves the original target pathway at initialization while progressively injecting reference-coordinate information across scales.

Tri-axis Likelihood Field Decoder. The conditioned features are decoded by a dual-head Tri-axis Likelihood Field (TLF) Decoder. As shown in Figure 2, the likelihood branch predicts $\mathcal{H} = \{H_x, H_y, H_z, H_o\}$, representing the dense likelihoods of the three projected axes and their common origin. Rather than collapsing each axis into a single deterministic prediction, these fields preserve distributed evidence and multiple plausible axis responses. As illustrated in Figure 3, weak texture or object symmetry may induce multiple plausible hypotheses for one or more axes.

In parallel, the geometry branch summarizes the conditioned bottleneck using spatial covariance pooling. Given pooled features $F \in \mathbb{R}^{C \times N}$, where $N = H_p W_p$ denotes the number of pooled spatial locations, we center each location across channels as $\bar{F} = \left(I_C - \frac{1}{C}\mathbf{1}\mathbf{1}^\top\right) F$. We then compute the spatial covariance $\Sigma = \frac{\bar{F}^\top \bar{F}}{C-1} + \epsilon I_N \in \mathbb{R}^{N \times N}$, where $\epsilon = 10^{-4}$ ensures numerical stability. The upper-triangular entries of Σ are vectorized and mapped by an MLP to $\mathcal{G} = \{c^G, d_x^G, d_y^G, d_z^G\}$, where $c^G \in [0,1]^2$ denotes the normalized image-plane origin and each $d_i^G \in \mathbb{R}^2$ is normalized to unit length. Unlike global average pooling, spatial covariance retains pairwise dependencies between image locations and thus provides complementary global structural guidance. Thus, $\mathcal{H}$ retains local and potentially ambiguous axis evidence, while $\mathcal{G}$ provides global structural guidance.

3.4 Geometry-Grounded Pose Recovery

The predicted likelihood fields encode plausible target-view axes but do not directly define a valid 3D rotation. Moreover, weak texture, symmetry, and occlusion may introduce multiple plausible axis configurations (Figure 3). We therefore introduce Geometry-Grounded Pose Recovery (GGPR), consisting of Tri-Axis Constrained Back-Projection (TACB) for pose-hypothesis generation and Hypothesis-Aware Axis Selection (HAAS) for ambiguity resolution.

Tri-Axis Constrained Back-Projection. Let $\mathbf{c}_\mathcal{H}$ denote the origin decoded from H_o. For each axis field H_i, $i \in \{x, y, z\}$, we extract several angularly distinct peaks $\mathbf{z}_i$ and define their unit directions as $\mathbf{d}_i = (\mathbf{z}_i - \mathbf{c}_\mathcal{H})/\|\mathbf{z}_i - \mathbf{c}_\mathcal{H}\|_2$. Candidate directions are retained according to their peak response, ray-wise likelihood, and agreement with the global direction $\mathbf{d}_i^\mathcal{G}$. Their combinations form multiple 2D tri-axis hypotheses.

Since the likelihood fields provide directions rather than metric endpoints, we additionally sample several axis extents ρ and obtain $\mathbf{p}_i = \mathbf{c}_\mathcal{H} + \rho \mathbf{d}_i$. For an image point $\mathbf{u}$, let $\mathbf{q}(\mathbf{u})$ denote its normalized camera ray obtained using $\mathbf{K}_t^{-1}$. Following the C2 formulation Zou et al. (2026a), TACB back-

projects the origin and axis endpoints onto their corresponding rays and solves the ray scales such that the recovered 3D axes are mutually orthogonal:

$$\mathbf{v}_i^h = \lambda_i^h \mathbf{q}(\mathbf{p}_i^h) - \lambda_o \mathbf{q}(\mathbf{c}_\mathcal{H}), \qquad (\mathbf{v}_i^h)^\top \mathbf{v}_j^h = 0, \quad i \neq j. \tag{1}$$

Here, λ_o fixes an arbitrary reconstruction scale and λ_i^h are obtained algebraically from the orthogonality constraints. Different direction combinations, extents, and algebraic branches produce a hypothesis set indexed by h. We form $\widetilde{\mathbf{R}}^h = [\bar{\mathbf{v}}_x^h, \bar{\mathbf{v}}_y^h, \bar{\mathbf{v}}_z^h]$, where $\bar{\mathbf{v}}_i^h$ is the normalized 3D axis, and obtain a proper rotation as $\mathbf{R}^h = \mathrm{Proj}_{\mathrm{SO}(3)}(\mathrm{Re}(\widetilde{\mathbf{R}}^h))$. The corresponding auxiliary translation is $\mathbf{t}^h = \mathrm{Re}(\lambda_o \mathbf{q}(\mathbf{c}_\mathcal{H}))$. TACB thus converts ambiguous 2D axis evidence into geometrically constrained pose hypotheses $\mathcal{P}^h = (\mathbf{R}^h, \mathbf{t}^h)$ (only $\mathbf{R}^h$ is retained as the final task output). More details are provided in Appendix A.2.

Hypothesis-Aware Axis Selection. Orthogonality alone does not resolve all hypotheses, since different image-space axis combinations may yield valid 3D rotations. HAAS therefore jointly evaluates projection consistency, TLF support, and algebraic stability. For each hypothesis h, let $\mathbf{X}^h = [\mathbf{X}_0^h, \mathbf{X}_x^h, \mathbf{X}_y^h, \mathbf{X}_z^h]$ denote the algebraic 3D point configuration returned by TACB and let $\widetilde{\mathbf{R}}^h$ denote its pre-projection axis matrix. We retain in $\mathcal{V}$ only finite and reprojectable hypotheses, while recording their imaginary residuals as an algebraic stability penalty. For each $h \in \mathcal{V}$, we reproject a canonical length-ℓ tri-axis using $(\mathbf{R}^h, \mathbf{t}^h)$, obtaining the origin $\widehat{\mathbf{c}}^h$ and axis directions $\widehat{\mathbf{d}}_i^h$. Its geometric inconsistency is

$$E_{\mathrm{geo}}^h = \frac{180}{3\pi} \sum_{i \in \{x,y,z\}} \arccos\Big(\mathrm{clip}\Big((\widehat{\mathbf{d}}_i^h)^\top \mathbf{d}_i^h, -1, 1\Big)\Big). \tag{2}$$

This term favors hypotheses whose recovered orthogonal frame reproduces the selected image-space directions. Let $\mathbf{z}_i^h$ be the peak selected for axis i by hypothesis h, with $\mathbf{d}_i^h = (\mathbf{z}_i^h - \mathbf{c}_\mathcal{H})/\|\mathbf{z}_i^h - \mathbf{c}_\mathcal{H}\|_2$. We measure its TLF support as $C_{\mathrm{TLF}}^h = \sum_i [\eta_r \mathcal{R}(H_i; \mathbf{c}_\mathcal{H}, \mathbf{d}_i^h) + \eta_p H_i(\mathbf{z}_i^h) + \eta_g (1 + (\mathbf{d}_i^h)^\top \mathbf{d}_i^\mathcal{G})/2]$, where $\mathcal{R}(\cdot)$ denotes the mean likelihood sampled along the corresponding axis ray. Algebraic instability is penalized by $E_{\mathrm{alg}}^h = \|\,\mathrm{Im}(\mathbf{X}^h)\|_\infty + \|\,\mathrm{Im}(\widetilde{\mathbf{R}}^h)\|_\infty$. HAAS ranks

$$h^\star = \arg\min_{h \in \mathcal{V}} \left(E_{\mathrm{geo}}^h + \beta E_{\mathrm{alg}}^h - C_{\mathrm{TLF}}^h\right), \qquad \widehat{\mathbf{R}}_t = \mathbf{R}^{h^\star}. \tag{3}$$

We use $\beta = 0.05$. As shown in Figure 3, HAAS selects the hypothesis that is projection-consistent and supported by both local TLF evidence and global axis geometry. More details are provided in Appendix A.3.

3.5 Training Objectives

AxisPose++ is trained entirely through the target-view tri-axis representation. We jointly supervise the dense likelihood fields $\mathcal{H}$ and the global axis geometry $\mathcal{G}$ predicted by the TLF Decoder.

Tri-axis likelihood supervision. Let $\widehat{H}_{b,i}$ and $H_{b,i}^*$ denote the predicted and ground-truth likelihood fields for axis $i \in \{x, y, z\}$. Since projected axes occupy only a small fraction of the image, we combine foreground-weighted binary cross-entropy with soft Dice loss. Each pixel is assigned $w_{b,i}(\mathbf{p}) = w_{\mathrm{fg}}$ if $H_{b,i}^*(\mathbf{p}) > \tau$ and w_{bg} otherwise:

$$\mathcal{L}_{\mathrm{WBCE}} = \tfrac{1}{Z} \sum_{b,i,\mathbf{p}} w_{b,i}(\mathbf{p})\, \mathrm{BCE}(\widehat{H}_{b,i}(\mathbf{p}), H_{b,i}^*(\mathbf{p})), \quad \mathcal{L}_{\mathrm{dice}} = \tfrac{1}{3B} \sum_{b,i} \left[1 - \tfrac{2\langle \widehat{H}_{b,i}, H_{b,i}^* \rangle + \epsilon}{\|\widehat{H}_{b,i}\|_1 + \|H_{b,i}^*\|_1 + \epsilon}\right]. \tag{4}$$

where $Z = \sum_{b,i,\mathbf{p}} w_{b,i}(\mathbf{p})$ and $\mathcal{L}_{\mathrm{axis}} = \mathcal{L}_{\mathrm{WBCE}} + \lambda_{\mathrm{dice}} \mathcal{L}_{\mathrm{dice}}$. The origin field H_o is supervised by MSE to localize the shared tri-axis origin.

Global geometry supervision. Complementary to the dense likelihood fields, the geometry branch predicts a normalized image-plane origin $\widehat{\mathbf{c}}^\mathcal{G}$ and three unit image-plane directions $\{\widehat{\mathbf{d}}_i^\mathcal{G}\}_{i \in \{x,y,z\}}$ to provide global structural guidance:

$$\mathcal{L}_{\mathrm{dir}} = \frac{1}{3} \sum_{i \in \{x,y,z\}} \left(1 - (\widehat{\mathbf{d}}_i^\mathcal{G})^\top \mathbf{d}_i^*\right). \tag{5}$$

The global origin is supervised with a Smooth-ℓ_1 loss $\mathcal{L}_{\mathrm{center}}$ against the normalized ground-truth image-plane origin $\mathbf{c}^*$ (this auxiliary loss does not supervise target translation). Finally, $\mathcal{L}_{\mathrm{total}} = \lambda_{\mathrm{axis}} \mathcal{L}_{\mathrm{axis}} + \lambda_{\mathrm{origin}} \mathcal{L}_{\mathrm{origin}} + \lambda_{\mathrm{center}} \mathcal{L}_{\mathrm{center}} + \lambda_{\mathrm{dir}} \mathcal{L}_{\mathrm{dir}}$. Loss settings are provided in Appendix A.4.

Table 1: Quantitative comparisons on ShapeNet, NAVI, and Objectron. **Bold** and underlined values denote the best and second-best results. OnePose++ and BoxDreamer use 48 and 5 reference views.

ShapeNet

	Acc@30 (%) ↑							Median angular error (°) ↓						
Method	Bus	Clock	Mug	Washer	Pistol	Train	Mean	Bus	Clock	Mug	Washer	Pistol	Train	Mean
OnePose++	54.5	48.9	37.2	62.9	57.2	37.1	49.6	24.5	39.1	49.9	19.5	28.7	85.4	41.2
BoxDreamer	30.2	24.8	28.8	43.0	39.9	20.3	31.2	76.9	80.1	70.0	42.7	66.7	82.2	69.8
RelPose	44.8	45.9	45.8	43.8	45.7	45.3	45.2	31.5	31.2	31.1	31.8	31.2	31.5	31.4
RelPose++	42.3	48.9	45.5	44.3	45.8	42.4	44.9	33.9	30.4	31.8	32.8	31.8	33.3	32.3
PIZZA	44.1	44.9	45.0	42.6	45.2	43.0	44.1	32.6	33.5	32.4	40.7	33.3	35.8	34.7
MicKey	49.4	46.5	45.8	42.2	45.4	47.0	46.1	30.1	31.3	31.5	32.4	31.4	31.0	31.3
Free3D	39.2	37.8	43.1	36.7	40.3	37.1	39.0	56.2	84.7	47.4	76.5	44.0	64.4	62.2
NOPE	48.8	49.0	46.3	59.4	57.9	53.7	52.5	34.2	33.1	32.3	23.7	18.2	27.9	28.2
OrienPose	58.7	57.1	**51.9**	65.5	62.6	58.9	59.1	18.3	25.4	**28.6**	14.5	**11.7**	22.8	20.2
AxisPose++	**78.2**	**67.0**	31.2	**79.2**	**78.0**	**75.3**	**68.2**	**6.6**	**15.1**	45.2	**3.6**	**11.7**	**10.3**	**15.4**

NAVI

Method	Bull	Bunny	Sink	Shark	School Bus	Mean	Bull	Bunny	Sink	Shark	School Bus	Mean
OnePose++	45.0	40.1	**49.3**	20.9	61.8	43.4	35.2	46.8	30.8	79.5	**21.1**	42.7
BoxDreamer	19.8	20.1	29.5	21.6	24.3	23.1	99.4	92.7	84.2	86.1	95.9	91.7
RelPose	**45.5**	42.7	43.8	39.0	54.1	45.0	**31.4**	32.0	30.7	34.3	28.2	**31.3**
RelPose++	28.9	28.3	27.7	42.9	38.7	33.3	42.2	40.4	41.4	31.9	35.6	38.3
PIZZA	37.6	49.5	47.5	39.6	57.4	46.3	59.6	35.1	**30.4**	32.6	27.5	37.0
MicKey	27.7	37.6	31.6	34.6	51.4	36.6	42.4	45.9	39.8	34.5	29.4	38.4
Free3D	26.9	35.7	27.2	30.1	35.8	31.1	72.1	56.2	86.4	69.3	79.2	72.6
NOPE	25.7	39.6	27.7	42.5	48.5	36.8	54.6	37.9	49.7	36.4	67.2	49.2
OrienPose	41.2	**56.4**	38.4	**55.9**	62.4	**50.9**	54.1	**26.9**	46.3	**28.5**	23.1	35.8
AxisPose++	30.7	48.8	29.1	50.4	**76.4**	47.1	39.4	30.1	45.8	29.2	22.7	33.5

Objectron

Method	Camera	Chair	Laptop	Book	Bottle	Cereal	Cup	Shoe	Mean	Camera	Chair	Laptop	Book	Bottle	Cereal	Cup	Shoe	Mean
OnePose++	59.0	69.1	53.5	46.9	33.4	52.2	42.8	43.7	50.1	21.4	**17.7**	28.9	32.6	68.1	29.4	57.8	35.3	36.4
BoxDreamer	41.9	22.9	21.8	49.2	31.6	27.5	26.1	26.3	30.9	39.7	45.1	87.5	35.3	37.2	74.8	69.3	48.6	54.7
RelPose	57.9	72.2	53.4	56.9	39.2	42.4	57.2	51.6	53.9	28.1	23.9	28.8	24.2	45.7	33.5	28.6	29.2	30.3
RelPose++	62.3	**83.6**	**68.2**	**73.7**	44.3	43.5	**64.5**	**64.7**	**63.1**	26.4	22.3	**24.3**	19.9	31.9	31.3	**26.5**	**25.9**	**26.1**
PIZZA	54.7	54.1	57.9	45.3	49.7	48.7	49.0	46.7	50.8	29.5	29.7	28.8	31.1	30.0	30.1	30.1	30.6	30.0
MicKey	18.6	10.8	18.2	66.0	20.3	10.3	22.7	19.5	23.3	41.6	45.9	44.1	**19.4**	36.3	39.8	36.8	41.2	38.1
Free3D	50.1	49.6	35.2	41.8	31.6	44.1	39.9	34.3	40.8	29.8	35.7	50.6	51.2	78.5	49.4	53.4	61.1	51.2
NOPE	58.1	51.8	54.1	44.7	46.0	49.3	48.7	53.5	50.8	28.0	29.8	28.7	32.2	30.5	30.2	30.2	29.2	29.9
OrienPose	61.8	56.2	56.5	49.3	45.5	55.0	49.5	54.0	53.5	27.2	28.9	28.5	30.1	30.6	29.1	30.1	28.3	29.1
AxisPose++	**72.4**	49.8	44.2	55.4	**65.1**	**60.3**	31.1	52.7	53.9	**20.9**	30.0	32.6	24.8	**26.1**	**23.5**	51.5	29.0	29.8

4 EXPERIMENT

Implementation Details. AxisPose++ is trained on a single RTX 4090 with a frozen DINOv2 ViT-S/14-reg encoder and a residual U-Net. We use AdamW for 100 epochs with batch size 24 and learning rate 10^{-4}. At inference, TACB retains five candidates per axis and evaluates six axis extents before HAAS selection. Full settings are provided in Appendix A.5.

Datasets and Metrics. We train on the NOPE ShapeNet split Nguyen et al. (2024a), containing 11 training and 6 unseen test categories. We further evaluate on NAVI Jampani et al. (2023) and Objectron Ahmadyan et al. (2021). Following prior protocols Nguyen et al. (2024a); Liu et al. (2026), we report median rotation error and ACC@30 for pairs within $45°$ relative rotation. Dataset statistics are provided in Appendix A.5.

Comparison Methods. We compare with nine RGB-only generalizable baselines: generate-and-compare methods NOPE Nguyen et al. (2024a), OrienPose Liu et al. (2026), and a Free3D-based variant Zheng & Vedaldi (2024); direct regression PIZZA Du et al. (2022); keypoint matching MicKey Barroso-Laguna et al. (2024); probabilistic relative-pose estimation RelPose Zhang et al. (2022) and RelPose++ Lin et al. (2024a); and multi-view geometry methods OnePose++ He et al. (2022) and BoxDreamer Yu et al. (2025). We use official pretrained models where available and do not fine-tune competitors on unseen objects. RelPose/RelPose++ use one reference image, whereas OnePose++/BoxDreamer use 48/5 references, following OrienPose Liu et al. (2026).

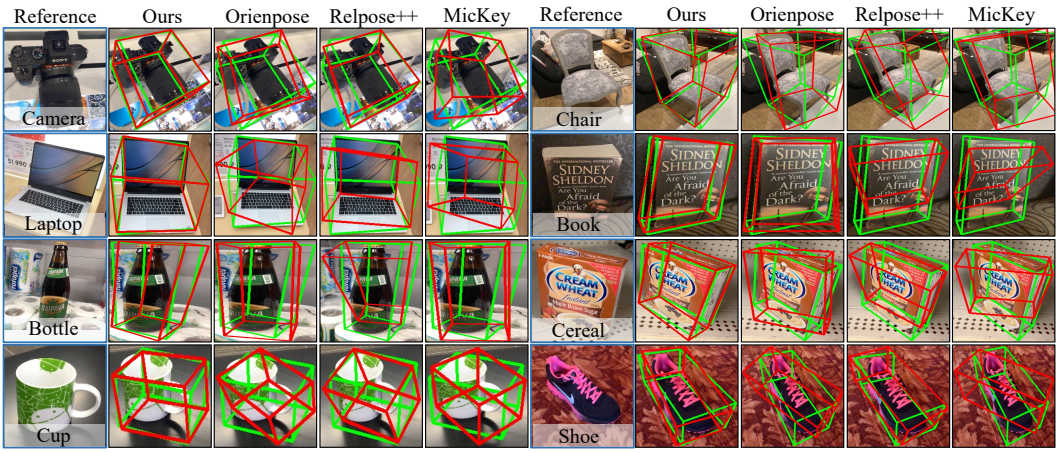

Figure 4: Qualitative comparisons on Objectron. Red and green boxes denote the prediction and ground truth, respectively.

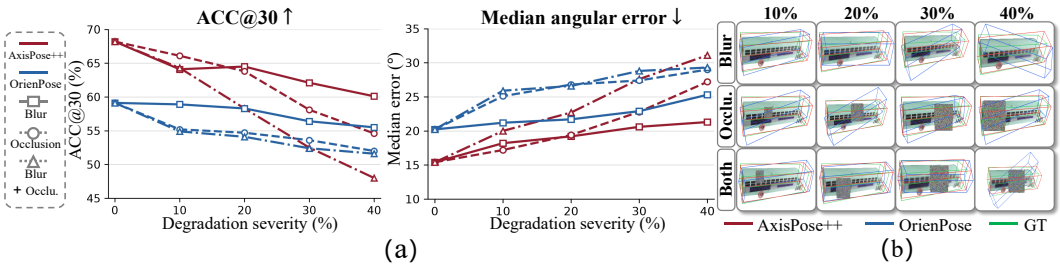

Figure 5: **Robustness to image degradation on ShapeNet.** (a) Acc@30 and median angular error under increasing blur, occlusion, and their combination. (b) Qualitative results under the same settings. Red, blue, and green denote AxisPose++, OrienPose, and the ground truth, respectively.

4.1 Comparison with Existing Methods

Quantitative Comparison. Table 1 compares AxisPose++ with existing methods on ShapeNet, NAVI, and Objectron. On six unseen ShapeNet categories, AxisPose++ achieves 68.2% mean Acc@30 and 15.4° median error, outperforming OrienPose by 9.1 points in Acc@30 and reducing the median error by 4.8°. It achieves the best Acc@30 on five of six categories, showing strong unseen-category generalization. Without real-world fine-tuning, AxisPose++ also transfers competitively to NAVI and Objectron. It attains 47.1%/ 33.5° on NAVI and 53.9%/ 29.8° on Objectron in Acc@30/median error, outperforming OrienPose in NAVI median error and slightly in Objectron Acc@30. Overall, the results demonstrate strong synthetic-to-real generalization while preserving competitive performance across real-world benchmarks.

Qualitative Results. Figure 4 shows qualitative comparisons on Objectron. AxisPose++ yields more geometrically consistent orientations across diverse real-world objects, including challenging cases with object symmetry and weak texture. These results indicate effective synthetic-to-real generalization of reference-anchored tri-axis transport. Additional results are provided in Appendix A.6.

Efficiency Comparison. As shown in Figure 1 (f), AxisPose++ (2.75 FPS) achieves 68.2% mean Acc@30 while running 43× faster than OrienPose (0.063 FPS) on ShapeNet, owing to the removal of per-query NVS and dense matching. AxisPose++ contains only 35.31M trainable parameters.

Robustness Analysis. Figure 5 evaluates robustness to blur, occlusion, and their combination. In (a), AxisPose++ generally maintains higher Acc@30 and lower median angular error than OrienPose under blur and moderate occlusion, indicating that tri-axis transport remains reliable when local appearance cues degrade. Under severe combined degradation, however, performance drops sharply as both appearance and structural evidence are corrupted. The qualitative results in Figure 5 (b)

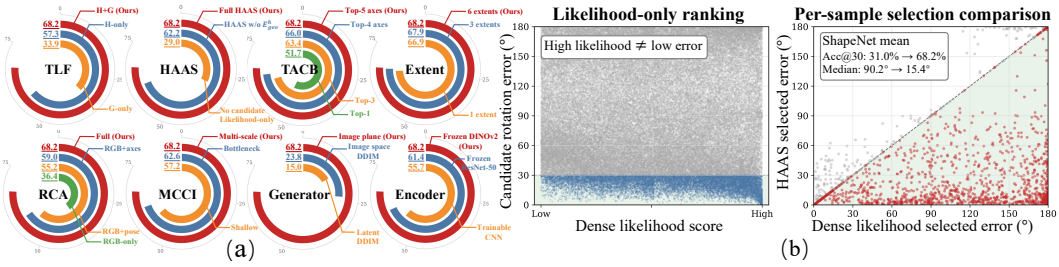

Figure 6: **Ablation analysis on ShapeNet.** (a) Mean Acc@30 (%) for eight design choices. (b) Dense likelihood alone cannot reliably rank TACB candidates (left); HAAS selects lower-error candidates on most samples (right).

exhibit the same trend, showing stable pose recovery under moderate degradation but failure when the observable object structure becomes highly incomplete.

4.2 Ablation Studies

Ablation of Eight Design Choices. Figure 6 (a) evaluates eight design choices of AxisPose++. **TLF:** Combining the dense likelihood fields $\mathcal{H}$ with global geometry $\mathcal{G}$ achieves 68.2% Acc@30, outperforming $\mathcal{H}$-only (57.3%) and $\mathcal{G}$-only (33.9%), demonstrating their complementary local and global cues. **HAAS:** Full HAAS reaches 68.2%, compared with 62.2% without E^h_{geo} and 29.0% when directly selecting the likelihood-only solution without candidate reasoning, confirming the importance of multi-hypothesis reasoning and geometric consistency. **TACB:** Increasing the retained axis directions from one to five steadily improves Acc@30 from 51.7% to 68.2%. **Extent:** Searching one, three, and six axis extents yields 66.9%, 67.9%, and 68.2%, showing a modest but consistent benefit from extent marginalization. **RCA:** RGB alone obtains 36.4%; adding the reference pose or projected axes improves it to 55.2% and 59.0%, while their joint use reaches 68.2%, validating their complementary anchoring roles. **MCCI:** Multi-scale injection (68.2%) outperforms bottleneck-only (62.6%) and shallow-only injection (57.2%). **Generator:** Direct image-plane prediction achieves 68.2%, outperforming the two generative alternatives (23.8% and 15.0%). **Encoder:** Frozen DINOv2 achieves 68.2%, compared with 61.4% for frozen ResNet-50 and 55.7% for a trainable CNN. Qualitative results are provided in Appendix A.7.

Candidate-ranking Analysis. Figure 6 (b) examines candidate selection within the TACB hypothesis pool. The left plot shows that high likelihood does not necessarily correspond to low rotation error, making likelihood alone unreliable for ranking pose candidates. The per-sample comparison on the right shows that HAAS selects lower-error hypotheses for most samples, improving Acc@30 from 31.0% to 68.2% and reducing the median error from 90.2° to 15.4°. Here, likelihood-only ranking selects the highest-supported hypothesis from the same TACB pool and is therefore distinct from the no-candidate likelihood-only baseline in Figure 6 (a) (HAAS).

Failure Case Analysis. As shown in Figure 7, weak texture and symmetry create indistinguishable orientations, causing HAAS to select an incorrect yet geometrically valid hypothesis (a). Severe occlusion and degradation may exclude the correct rotation from the TACB candidate pool (b).

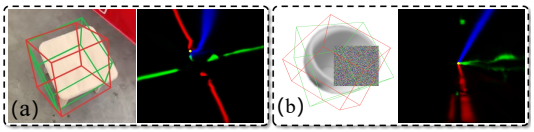

Figure 7: Failure cases caused by intrinsic ambiguity (a) and severe appearance degradation (b).

5 Conclusion

We present AxisPose++, a reference-anchored framework for single-reference unseen object rotation estimation. It transports a compact tri-axis structure from a posed reference to the target view and resolves ambiguous axis predictions through geometry-grounded recovery. This formulation avoids per-query novel-view synthesis, depth prediction, and explicit cross-view matching. Experiments demonstrate strong generalization and an effective accuracy-efficiency trade-off.

AI USE STATEMENT

Generative AI tools were used to assist with language editing, figure refinement, and feedback on the presentation of the methodology and experiments. All AI-assisted content was reviewed and verified by the authors, who take full responsibility for the final content of this paper.

ETHICS STATEMENT

This work studies single-reference 3D object rotation estimation. We do not collect new human-subject data or release personal data. Our experiments use rendered ShapeNet data and established public benchmarks, following their original terms of use.

REPRODUCIBILITY STATEMENT

To support reproducibility, we provide an anonymous repository containing the complete implementation, training and evaluation configurations, pretrained ShapeNet weights, and a ready-to-run ShapeNet evaluation subset. The repository documents environment setup, training, inference, evaluation metrics, and visualization. The full ShapeNet data can be obtained from the official NOPE resources and evaluated using the documented data layout. The paper and appendix further specify the model architecture, loss weights, training schedule, inference hyperparameters, datasets, and evaluation protocol.

REFERENCES

Adel Ahmadyan, Liangkai Zhang, Artsiom Ablavatski, Jianing Wei, and Matthias Grundmann. Objectron: A large scale dataset of object-centric videos in the wild with pose annotations. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7818–7827. IEEE, 2021.

Axel Barroso-Laguna, Sowmya Munukutla, Victor Adrian Prisacariu, and Eric Brachmann. Matching 2d images in 3d: Metric relative pose from metric correspondences. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4852–4863, 2024.

Dengsheng Chen, Jun Li, Zheng Wang, and Kai Xu. Learning canonical shape space for category-level 6d object pose and size estimation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11970–11979. IEEE, 2020.

Jianqi Chen, Biao Zhang, Xiangjun Tang, and Peter Wonka. Posegam: Robust unseen object pose estimation via geometry-aware multi-view reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7197–7208, 2026.

Yuming Du, Yang Xiao, Michael Ramamonjisoa, Vincent Lepetit, et al. Pizza: A powerful image-only zero-shot zero-cad approach to 6 dof tracking. In *2022 International Conference on 3D Vision (3DV)*, pp. 515–525. IEEE, 2022.

Amir Hamza, Davide Boscaini, Weihang Li, Benjamin Busam, and Fabio Poiesi. Generative 6d pose estimation via conditional flow matching. *arXiv preprint arXiv:2602.19719*, 2026.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Xingyi He, Jiaming Sun, Yuang Wang, Di Huang, Hujun Bao, and Xiaowei Zhou. Onepose++: Keypoint-free one-shot object pose estimation without cad models. *Advances in Neural Information Processing Systems*, 35:35103–35115, 2022.

Varun Jampani, Kevis-Kokitsi Maninis, Andreas Engelhardt, Arjun Karpur, Karen Truong, Kyle Sargent, Stefan Popov, André Araujo, Ricardo Martin Brualla, Kaushal Patel, et al. Navi: Category-agnostic image collections with high-quality 3d shape and pose annotations. *Advances in Neural Information Processing Systems*, 36:76061–76084, 2023.

Taeyeop Lee, Bowen Wen, Minjun Kang, Gyuree Kang, In So Kweon, and Kuk-Jin Yoon. Any6d: Model-free 6d pose estimation of novel objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11633–11643, 2025.

Weihang Li, Hongli Xu, Junwen Huang, Hyunjun Jung, Peter KT Yu, Nassir Navab, and Benjamin Busam. Gce-pose: Global context enhancement for category-level object pose estimation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 27154–27165, 2025.

Amy Lin, Jason Y Zhang, Deva Ramanan, and Shubham Tulsiani. Relpose++: Recovering 6d poses from sparse-view observations. In *2024 International Conference on 3D Vision (3DV)*, pp. 106–115. IEEE, 2024a.

Jiehong Lin, Zewei Wei, Yabin Zhang, and Kui Jia. Vi-net: Boosting category-level 6d object pose estimation via learning decoupled rotations on the spherical representations. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 14001–14011, 2023.

Jiehong Lin, Lihua Liu, Dekun Lu, and Kui Jia. Sam-6d: Segment anything model meets zero-shot 6d object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 27906–27916, 2024b.

Jian Liu, Wei Sun, Hui Yang, Zhiwen Zeng, Chongpei Liu, Jin Zheng, Xingyu Liu, Hossein Rahmani, Nicu Sebe, and Ajmal Mian. Deep learning-based object pose estimation: A comprehensive survey. *arXiv preprint arXiv:2405.07801*, 2024.

Jianhui Liu, Yukang Chen, Xiaoqing Ye, and Xiaojuan Qi. Ist-net: Prior-free category-level pose estimation with implicit space transformation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13978–13988, 2023.

Yating Liu, Zhaoshuai Qi, Yang Zou, Yongnan Yang, Shizhou Zhang, and Yanning Zhang. Orienpose: Orientation-guided novel view synthesis for single-image unseen object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26813–26823, 2026.

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. In *European Conference on Computer Vision*, pp. 298–315. Springer, 2022.

Sungphill Moon, Hyeontae Son, Dongcheol Hur, and Sangwook Kim. Co-op: Correspondence-based novel object pose estimation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 11622–11632, 2025.

Van Nguyen Nguyen, Thibault Groueix, Georgy Ponimatkin, Yinlin Hu, Renaud Marlet, Mathieu Salzmann, and Vincent Lepetit. Nope: Novel object pose estimation from a single image. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 17923–17932, 2024a.

Van Nguyen Nguyen, Thibault Groueix, Mathieu Salzmann, and Vincent Lepetit. Gigapose: Fast and robust novel object pose estimation via one correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9903–9913, 2024b.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Panwang Pan, Zhiwen Fan, Brandon Y Feng, Peihao Wang, Chenxin Li, and Zhangyang Wang. Learning to estimate 6dof pose from limited data: A few-shot, generalizable approach using rgb images. In *2024 International Conference on 3D Vision (3DV)*, pp. 1059–1071. IEEE, 2024.

Zhaoshuai Qi, Yating Liu, and Yanning Zhang. Matching quality-guided model-free satellite pose estimation. *Engineering Applications of Artificial Intelligence*, 161:112194, 2025.

Mahdi Rad and Vincent Lepetit. Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth. In *2017 IEEE international conference on computer vision (ICCV)*, pp. 3848–3856. IEEE, 2017.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Jiaming Sun, Zihao Wang, Siyu Zhang, Xingyi He, Hongcheng Zhao, Guofeng Zhang, and Xiaowei Zhou. Onepose: One-shot object pose estimation without cad models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6825–6834, 2022.

Meng Tian, Marcelo H Ang Jr, and Gim Hee Lee. Shape prior deformation for categorical 6d object pose and size estimation. In *European Conference on Computer Vision*, pp. 530–546. Springer, 2020.

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 *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2637–2646. IEEE, 2019.

Jiahui Wang, Haiyue Zhu, Haoren Guo, Abdullah Al Mamun, Cheng Xiang, and Tong Heng Lee. Singref6d: Monocular novel object pose estimation with a single rgb reference. *Advances in Neural Information Processing Systems*, 38:25400–25437, 2026.

Bowen Wen, Jonathan Tremblay, Valts Blukis, Stephen Tyree, Thomas Müller, Alex Evans, Dieter Fox, Jan Kautz, and Stan Birchfield. Bundlesdf: Neural 6-dof tracking and 3d reconstruction of unknown objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 606–617, 2023.

Bowen Wen, Wei Yang, Jan Kautz, and Stan Birchfield. Foundationpose: Unified 6d pose estimation and tracking of novel objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 17868–17879, 2024.

Yuanhong Yu, Xingyi He, Chen Zhao, Junhao Yu, Jiaqi Yang, Ruizhen Hu, Yujun Shen, Xing Zhu, Xiaowei Zhou, and Sida Peng. Boxdreamer: Dreaming box corners for generalizable object pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9374–9384, 2025.

Jason Y Zhang, Deva Ramanan, and Shubham Tulsiani. Relpose: Predicting probabilistic relative rotation for single objects in the wild. In *European Conference on Computer Vision*, pp. 592–611. Springer, 2022.

Chen Zhao, Yinlin Hu, and Mathieu Salzmann. Locposenet: Robust location prior for unseen object pose estimation. In *2024 International Conference on 3D Vision (3DV)*, pp. 1072–1081. IEEE, 2024.

Chuanxia Zheng and Andrea Vedaldi. Free3d: Consistent novel view synthesis without 3d representation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9720–9731. IEEE, 2024.

Yang Zou, Wenyou Chang, Yating Liu, Yifeng Hao, Rui Hu, Jiaqi Yang, and Zhaoshuai Qi. Indoor 3d reconstruction using an unknown camera-projector pair. *Optics Express*, 34(13):24745–24767, 2026a.

Yang Zou, Zhaoshuai Qi, Yating Liu, Weiyi Liu, Zihao Xu, Weipeng Sun, Xingyuan Li, Jiaqi Yang, and Yanning Zhang. Axispose: Model-free matching-free single-shot 6d object pose estimation via axis generation. *IEEE Transactions on Circuits and Systems for Video Technology*, 2026b.

A APPENDIX

A.1 OVERVIEW

This appendix provides additional methodological details, implementation settings, and experimental evidence for AxisPose++. Appendix A.2 describes the relation to the preceding AxisPose formulation and presents the complete derivation of Tri-Axis Constrained Back-Projection. Appendix A.3 details the validity checks, scoring terms, and selection procedure used by Hypothesis-Aware Axis Selection. Appendix A.4 specifies the construction of the target tri-axis representation and the complete training objective, while Appendix A.5 reports the implementation, data preparation, and inference settings required for reproduction. Finally, Appendices A.6 and A.7 provide additional qualitative comparisons, component-wise ablations, and an analysis of the accuracy-efficiency trade-off.

A.2 TRI-AXIS CONSTRAINED BACK-PROJECTION

Relation to AxisPose. AxisPose Zou et al. (2026b) introduced the C2-based algebraic back-projection formulation for model-free single-shot cross-instance level pose estimation. Given one predicted 2D tri-axis configuration, C2 recovers a 3D orthogonal axis frame by constraining the corresponding camera rays. We adopt this algebraic solver as a geometric building block rather than claiming a new solver.

Our setting differs fundamentally from AxisPose. AxisPose predicts a single query-only 2D axis configuration, whereas AxisPose++ predicts reference-conditioned dense likelihood fields over target-view axis directions. TACB therefore lifts C2 from a deterministic inversion of one 2D triad to a multi-hypothesis inference procedure that combines multiple angular modes from each axis field, multiple image-space extents, and the algebraic C2 branches into a set of geometrically valid 3D pose hypotheses. The final pose is selected using reprojection consistency with the predicted axis fields and reference-anchored constraints. AxisPose is not included in the experiment because it addresses cross-instance pose estimation rather than the unseen-category single-reference generalization setting considered here.

Extracting 2D directional hypotheses. The target-view network predicts three dense likelihood fields $\{H_x, H_y, H_z\}$ and an origin heatmap H_o. We first decode the image-space origin $\mathbf{c}_{\mathcal{H}} = \arg\max H_o$. For each axis field H_i, where $i \in \{x, y, z\}$, we collect local maxima outside a small exclusion region around $\mathbf{c}_{\mathcal{H}}$. A peak at image location $\mathbf{z}_{i,k}$ induces a directed 2D ray

$$\mathbf{d}_{i,k} = \frac{\mathbf{z}_{i,k} - \mathbf{c}_{\mathcal{H}}}{\|\mathbf{z}_{i,k} - \mathbf{c}_{\mathcal{H}}\|_2}. \tag{6}$$

Its confidence combines the local peak value, the average response sampled along the corresponding ray, and its agreement with the global direction $\mathbf{d}_i^{\mathcal{G}}$ predicted by the geometry head

$$s_{i,k} = 0.6\, s_{i,k}^{\text{ray}} + 0.4\, s_{i,k}^{\text{peak}} + \gamma \frac{1 + \mathbf{d}_{i,k}^{\top} \mathbf{d}_i^{\mathcal{G}}}{2}, \tag{7}$$

where $\gamma = 0.5$. We retain angularly distinct directions using non-maximum suppression in direction space. In the official implementation, we retain five directions per axis with a minimum angular separation of 5°. Taking the Cartesian product of the retained x, y, and z directions produces $5^3 = 125$ 2D tri-axis hypotheses.

From image rays to a C2 configuration. For one 2D tri-axis hypothesis, let ρ be a sampled image-space axis extent and define

$$\mathbf{p}_i = \mathbf{c}_{\mathcal{H}} + \rho \mathbf{d}_i, \qquad i \in \{x, y, z\}. \tag{8}$$

We use the normalized camera ray

$$\mathbf{n}(\mathbf{u}) = \frac{\mathbf{K}_t^{-1}[u, v, 1]^{\top}}{\left\|\mathbf{K}_t^{-1}[u, v, 1]^{\top}\right\|_2}. \tag{9}$$

Let $\mathbf{n}_0 = \mathbf{n}(\mathbf{c}_{\mathcal{H}})$ and $\mathbf{n}_i = \mathbf{n}(\mathbf{p}_i)$. The origin has a fixed arbitrary reconstruction scale

$$\mathbf{X}_0 = \lambda_0 \mathbf{n}_0, \tag{10}$$

while the three axis endpoints lie on their respective camera rays

$$\mathbf{X}_i = \lambda_i \mathbf{n}_i, \qquad \mathbf{v}_i = \mathbf{X}_i - \mathbf{X}_0. \tag{11}$$

TACB enforces the C2 orthogonality constraints

$$\mathbf{v}_x^\top \mathbf{v}_y = \mathbf{v}_x^\top \mathbf{v}_z = \mathbf{v}_y^\top \mathbf{v}_z = 0. \tag{12}$$

Algebraic C2 solution. Let $A = \mathbf{n}_x^\top \mathbf{n}_y$, $B = \mathbf{n}_x^\top \mathbf{n}_z$, $C = \mathbf{n}_y^\top \mathbf{n}_z$, $D = \mathbf{n}_x^\top \mathbf{X}_0$, $E = \mathbf{n}_y^\top \mathbf{X}_0$, $F = \mathbf{n}_z^\top \mathbf{X}_0$, and $K_0 = \mathbf{X}_0^\top \mathbf{X}_0$. The first two constraints in Eq. (12) yield

$$\lambda_y = \frac{D\lambda_x - K_0}{A\lambda_x - E}, \qquad \lambda_z = \frac{D\lambda_x - K_0}{B\lambda_x - F}. \tag{13}$$

Substituting Eq. (13) into the remaining orthogonality constraint gives a quadratic equation

$$a\lambda_x^2 + b\lambda_x + c = 0, \tag{14}$$

where

$$\begin{aligned} a &= CD^2 - BDE - ADF + ABK_0, \\ b &= -2CDK_0 + (BK_0 + FD)E + (AK_0 + DE)F - (AF + BE)K_0, \\ c &= CK_0^2 - EFK_0. \end{aligned} \tag{15}$$

The two roots of Eq. (14) correspond to two algebraic C2 branches, each of which produces a 3D triad $\{\mathbf{v}_x, \mathbf{v}_y, \mathbf{v}_z\}$.

We normalize the recovered axes and form

$$\widetilde{\mathbf{R}}^h = \left[\frac{\mathbf{v}_x^h}{\|\mathbf{v}_x^h\|_2}, \frac{\mathbf{v}_y^h}{\|\mathbf{v}_y^h\|_2}, \frac{\mathbf{v}_z^h}{\|\mathbf{v}_z^h\|_2} \right]. \tag{16}$$

To remove numerical non-orthogonality, we project the real-valued matrix onto $\mathrm{SO}(3)$

$$\mathbf{R}^h = \mathrm{Proj}_{\mathrm{SO}(3)} \left(\mathrm{Re}(\widetilde{\mathbf{R}}^h) \right), \qquad \mathbf{t}^h = \mathrm{Re}(\mathbf{X}_0). \tag{17}$$

The projection is implemented with an SVD and a determinant correction. The final task output is the rotation $\mathbf{R}^h$.

Hypothesis pool and validity filtering. In the official implementation, we retain five candidates per axis, use six extents $\rho \in \{80, 100, 120, 140, 160, 180\}$, and evaluate both C2 branches, yielding $5^3 \times 6 \times 2 = 1500$ pose hypotheses per test pair. Non-finite or non-reprojectable hypotheses are discarded. We use real-valued C2 recovery while retaining imaginary residuals as an algebraic-stability cue. The remaining hypotheses are ranked by HAAS using directional reprojection consistency, TLF support, and algebraic stability, as detailed in Appendix A.3.

A.3 Hypothesis-Aware Axis Selection

Candidate validity and reprojection. TACB produces a set of C2 pose hypotheses $\mathcal{H}$. Each hypothesis $h \in \mathcal{H}$ contains an algebraic point configuration $\mathbf{X}^h = [\mathbf{X}_0^h, \mathbf{X}_x^h, \mathbf{X}_y^h, \mathbf{X}_z^h]$, a pre-projection axis matrix $\widetilde{\mathbf{R}}^h$, and the proper rotation $\mathbf{R}^h \in \mathrm{SO}(3)$ obtained after real-value recovery and orthogonal projection. HAAS first removes non-finite hypotheses and hypotheses whose recovered origin or canonical axis endpoints cannot be validly reprojected into the target image. The remaining set is denoted by $\mathcal{V}$.

For each valid hypothesis, HAAS reprojects the origin and a canonical length-ℓ 3D tri-axis. Let $\mathbf{e}_i$ denote the i-th canonical basis vector and let $\pi_{\mathbf{K}_t}(\cdot)$ denote perspective projection under the target intrinsics. The projected origin and projected axis endpoints are

$$\widehat{\mathbf{c}}^h = \pi_{\mathbf{K}_t}(\mathbf{t}^h), \qquad \widehat{\mathbf{p}}_i^h = \pi_{\mathbf{K}_t}(\mathbf{t}^h + \ell \mathbf{R}^h \mathbf{e}_i). \tag{18}$$

The induced projected directions are

$$\widehat{\mathbf{d}}_i^h = \frac{\widehat{\mathbf{p}}_i^h - \widehat{\mathbf{c}}^h}{\left\| \widehat{\mathbf{p}}_i^h - \widehat{\mathbf{c}}^h \right\|_2}. \tag{19}$$

The value of $\ell > 0$ is immaterial. The projected origin and normalized axis directions are invariant to the arbitrary positive C2 reconstruction scale.

We measure the directional reprojection inconsistency as

$$E^h_{\text{dir}} = \frac{180}{3\pi} \sum_{i\in\{x,y,z\}} \arccos\left(\text{clip}\left((\widehat{\mathbf{d}}^h_i)^\top \mathbf{d}^h_i, -1, 1\right)\right), \tag{20}$$

Here, $\mathbf{d}^h_i$ is the 2D axis direction from which hypothesis h was constructed. This reprojection test is important because C2 enforces orthogonality in 3D, whereas HAAS further verifies whether the resulting proper rotation remains consistent with the original target-view axis evidence.

TLF support and algebraic stability. Let $\mathbf{z}^h_i$ be the selected local peak for axis i in hypothesis h. Its ray direction is $\mathbf{d}^h_i = (\mathbf{z}^h_i - \mathbf{c}_\mathcal{H})/\|\mathbf{z}^h_i - \mathbf{c}_\mathcal{H}\|_2$. We define the target-view likelihood support of the full tri-axis hypothesis as

$$C^h_{\text{TLF}} = \sum_{i\in\{x,y,z\}} \left[0.6\,\mathcal{R}(H_i; \mathbf{c}_\mathcal{H}, \mathbf{d}^h_i) + 0.4\,H_i(\mathbf{z}^h_i) + 0.5\frac{1 + (\mathbf{d}^h_i)^\top \mathbf{d}^\mathcal{G}_i}{2}\right], \tag{21}$$

where $\mathcal{R}(H_i; \mathbf{c}_\mathcal{H}, \mathbf{d}^h_i)$ is the mean likelihood sampled along the ray beginning at $\mathbf{c}_\mathcal{H}$ in direction $\mathbf{d}^h_i$, and $\mathbf{d}^\mathcal{G}_i$ is the global direction predicted by the geometry head. The first two terms measure dense local TLF evidence, while the last term measures agreement with the global geometry prediction.

C2 may produce complex-valued intermediate quantities for numerically unstable configurations. We retain the real-valued reconstruction used to form $\mathbf{R}^h$ and record the corresponding algebraic residual

$$E^h_{\text{alg}} = \left\|\text{Im}(\mathbf{X}^h)\right\|_\infty + \left\|\text{Im}(\widetilde{\mathbf{R}}^h)\right\|_\infty. \tag{22}$$

Thus, a hypothesis is not favoured merely because it yields an orthogonal frame. It must also be supported by the target likelihood fields and remain numerically stable under the C2 reconstruction. A common fixed C2 scale is used for all hypotheses when evaluating this algebraic residual.

Final hypothesis ranking. The final HAAS score is

$$E^h_{\text{HAAS}} = E^h_{\text{dir}} + 0.05E^h_{\text{alg}} - C^h_{\text{TLF}}. \tag{23}$$

HAAS selects the hypothesis with the lowest score

$$h^\star = \arg\min_{h\in\mathcal{V}} E^h_{\text{HAAS}}, \qquad \widehat{\mathbf{R}}_t = \mathbf{R}^{h^\star}. \tag{24}$$

The selection therefore favours a hypothesis whose recovered 3D axes reproduce the selected 2D rays, receive strong dense likelihood support, agree with the global axis prediction, and avoid algebraically unstable C2 solutions.

A.4 Training Objectives

Target tri-axis representation. The training targets are constructed entirely in the target image plane. Given the ground-truth target pose and camera intrinsics, we project the object origin and the three canonical axes onto the target image. The projected x, y, and z axes are rendered in red, green, and blue, respectively. For each axis, a binary mask is obtained by selecting pixels whose corresponding colour channel is dominant and exceeds a threshold of 0.3. Each mask is convolved with a 9×9 Gaussian kernel with standard deviation $\sigma = 2$, and the resulting response is normalized by its channel-wise maximum. This produces the three soft likelihood targets $\{H^*_{b,x}, H^*_{b,y}, H^*_{b,z}\}$. The ground-truth origin field $H^*_{b,o}$ is generated by placing a two-dimensional Gaussian with $\sigma = 2$ at the projected object origin.

Tri-axis likelihood loss. Let $\widehat{A}_{b,i}(\mathbf{p})$ denote the predicted logit at pixel $\mathbf{p}$ for axis $i \in \{x, y, z\}$, and let $\widehat{H}_{b,i}(\mathbf{p}) = \text{sigmoid}(\widehat{A}_{b,i}(\mathbf{p}))$. Because the rendered axes occupy only a small portion of the image, foreground pixels receive a larger weight

$$w_{b,i}(\mathbf{p}) = \begin{cases} w_{\text{fg}}, & H^*_{b,i}(\mathbf{p}) > \tau, \\ w_{\text{bg}}, & \text{otherwise.} \end{cases} \tag{25}$$

Algorithm 1. Hypothesis-Aware Axis Selection

Input C2 hypotheses $\mathcal{H}$, target likelihood fields $\{H_x, H_y, H_z\}$, target origin $\mathbf{c}_{\mathcal{H}}$, global directions $\{\mathbf{d}_i^{\mathcal{G}}\}$, and target intrinsics $\mathbf{K}_t$.

Output Selected target rotation $\widehat{\mathbf{R}}_t$.

```
Initialize V ← ∅.
for each hypothesis h ∈ H do
    if h is non-finite or cannot be validly reprojected then continue.
    Reproject t^h and t^h + ℓR^h e_i for i ∈ {x, y, z}.
    Compute E^h_dir using Eq.( 20).
    Compute C^h_TLF using Eq.( 21).
    Compute E^h_alg using Eq.( 22).
    Compute E^h_HAAS using Eq.( 23).
    Add h to V.
end for
h* ← arg min_{h∈V} E^h_HAAS.
return R̂_t ← R^{h*}.
```

The foreground-weighted binary cross-entropy loss is

$$\mathcal{L}_{\mathrm{WBCE}} = \frac{1}{Z}\sum_{b=1}^{B}\sum_{i\in\{x,y,z\}}\sum_{\mathbf{p}} w_{b,i}(\mathbf{p})\,\mathrm{BCE}\left(\widehat{H}_{b,i}(\mathbf{p}), H^*_{b,i}(\mathbf{p})\right), \qquad Z = \sum_{b,i,\mathbf{p}} w_{b,i}(\mathbf{p}). \tag{26}$$

In implementation, this term is evaluated directly from $\widehat{A}_{b,i}$ using the numerically stable binary-cross-entropy-with-logits operation. We further employ a soft Dice loss to encourage spatial overlap between each predicted likelihood field and its target

$$\mathcal{L}_{\mathrm{dice}} = \frac{1}{3B}\sum_{b=1}^{B}\sum_{i\in\{x,y,z\}}\left[1 - \frac{2\sum_{\mathbf{p}}\widehat{H}_{b,i}(\mathbf{p})H^*_{b,i}(\mathbf{p}) + \epsilon}{\sum_{\mathbf{p}}\widehat{H}_{b,i}(\mathbf{p}) + \sum_{\mathbf{p}}H^*_{b,i}(\mathbf{p}) + \epsilon}\right]. \tag{27}$$

The complete likelihood-field loss is

$$\mathcal{L}_{\mathrm{axis}} = \mathcal{L}_{\mathrm{WBCE}} + \lambda_{\mathrm{dice}}\mathcal{L}_{\mathrm{dice}}. \tag{28}$$

Origin likelihood loss. Let $\widehat{H}_{b,o}$ be the predicted origin likelihood field. Its supervision is given by the mean squared error

$$\mathcal{L}_{\mathrm{origin}} = \frac{1}{BHW}\sum_{b=1}^{B}\sum_{\mathbf{p}}\left(\widehat{H}_{b,o}(\mathbf{p}) - H^*_{b,o}(\mathbf{p})\right)^2. \tag{29}$$

This term only localizes the shared tri-axis origin in the target image and does not provide metric translation supervision.

Geometry-target extraction. To ensure that the global geometry branch and the dense likelihood fields share the same target representation, the global center and direction targets are derived from the ground-truth axis fields. For each $H^*_{b,i}$, we first compute its likelihood-weighted mean

$$\boldsymbol{\mu}_{b,i} = \frac{\sum_{\mathbf{p}} H^*_{b,i}(\mathbf{p})\mathbf{p}}{\sum_{\mathbf{p}} H^*_{b,i}(\mathbf{p}) + \epsilon} \tag{30}$$

and weighted covariance matrix

$$\boldsymbol{\Sigma}_{b,i} = \frac{\sum_{\mathbf{p}} H^*_{b,i}(\mathbf{p})\,(\mathbf{p}-\boldsymbol{\mu}_{b,i})\,(\mathbf{p}-\boldsymbol{\mu}_{b,i})^\top}{\sum_{\mathbf{p}} H^*_{b,i}(\mathbf{p}) + \epsilon}. \tag{31}$$

The eigenvector associated with the largest eigenvalue of $\boldsymbol{\Sigma}_{b,i}$ defines the unoriented image-plane axis line. We intersect all three pairs of fitted axis lines and combine their intersection points using weights proportional to the absolute sine of the angle between the corresponding directions. Nearly

parallel pairs therefore contribute less to the recovered center. The resulting pixel-space center is normalized as

$$\mathbf{c}_b^* = \left[\frac{c_{b,x}}{W-1}, \frac{c_{b,y}}{H-1} \right]^\top. \tag{32}$$

The sign ambiguity of each principal direction is resolved by orienting it from $\mathbf{c}_b^*$ toward its likelihood-weighted mean $\boldsymbol{\mu}_{b,i}$. The resulting unit direction is denoted by $\mathbf{d}_{b,i}^*$. This construction makes the auxiliary geometry targets consistent with the spatial structure encoded by the dense axis likelihood fields.

Global geometry loss. The geometry branch predicts a normalized image-plane center $\widehat{\mathbf{c}}_b^{\mathcal{G}}$ and three image-plane directions $\widehat{\mathbf{d}}_{b,i}^{\mathcal{G}}$. Both predicted and target directions are normalized before computing the cosine loss

$$\mathcal{L}_{\text{dir}} = \frac{1}{3B} \sum_{b=1}^{B} \sum_{i \in \{x,y,z\}} \left[1 - \left(\frac{\widehat{\mathbf{d}}_{b,i}^{\mathcal{G}}}{\|\widehat{\mathbf{d}}_{b,i}^{\mathcal{G}}\|_2 + \epsilon} \right)^\top \left(\frac{\mathbf{d}_{b,i}^*}{\|\mathbf{d}_{b,i}^*\|_2 + \epsilon} \right) \right]. \tag{33}$$

The normalized center is supervised using the Smooth-ℓ_1 loss

$$\mathcal{L}_{\text{center}} = \frac{1}{B} \sum_{b=1}^{B} \text{SmoothL1} \left(\widehat{\mathbf{c}}_b^{\mathcal{G}}, \mathbf{c}_b^* \right). \tag{34}$$

These geometry losses provide compact global structural guidance complementary to the spatially distributed TLF supervision.

Complete objective. The complete training objective is

$$\mathcal{L}_{\text{total}} = \lambda_{\text{axis}} \mathcal{L}_{\text{axis}} + \lambda_{\text{origin}} \mathcal{L}_{\text{origin}} + \lambda_{\text{center}} \mathcal{L}_{\text{center}} + \lambda_{\text{dir}} \mathcal{L}_{\text{dir}}. \tag{35}$$

For the experiments, we use

$$\begin{aligned} &\lambda_{\text{axis}} = \lambda_{\text{origin}} = \lambda_{\text{center}} = \lambda_{\text{dir}} = \lambda_{\text{dice}} = 1, \\ &w_{\text{fg}} = 20, \qquad w_{\text{bg}} = 1, \qquad \tau = 0.05, \qquad \epsilon = 10^{-6}. \end{aligned} \tag{36}$$

The stability threshold used when combining pairwise axis intersections is set to 0.05. No direct loss is imposed on the final three-dimensional rotation. Instead, the model learns target-view axis likelihoods and global image-plane geometry, which are subsequently converted into rotation hypotheses by TACB and ranked by HAAS.

A.5 Experimental Settings

Implementation Details. We implement AxisPose++ in PyTorch 2.3.0 and train it on a single NVIDIA GeForce RTX 4090 with CUDA 12.1. All reference and target images are resized to 336×336 and normalized using ImageNet statistics. The reference branch employs the official DINOv2 ViT-S/14-reg model. We extract its ninth-block features and freeze all pretrained parameters throughout training. The residual U-Net uses a base channel width of 64. In the global geometry branch, the bottleneck feature is adaptively pooled to 14×14, and the vectorized spatial covariance is processed by a two-layer MLP with a hidden dimension of 512 and no dropout. We train the model for 100 epochs using AdamW with a batch size of 24, a constant learning rate of 1×10^{-4}, weight decay of 1×10^{-4}, automatic mixed precision, and a gradient-norm threshold of 1.0. Validation is performed every two epochs, and the checkpoint with the lowest validation loss is used for evaluation. SingRef6D Wang et al. (2026) is omitted from the quantitative comparison because no official implementation or pretrained model is publicly available. The complete training objectives and their coefficients are provided in Appendix A.4.

Data Preparation. We use the same preprocessed ShapeNet training and test data, reference-target pairs, and pose annotations as OrienPose Liu et al. (2026). The training split contains 8,695 object instances and 154,834 reference-target pairs, while the unseen-category test split contains 561 instances and 9,986 pairs. The real-world evaluation contains 635 NAVI pairs and 5,866 Objectron pairs covering 1,124 object instances. The dataset composition, category splits, and evaluation metrics follow the protocol described in the main paper.

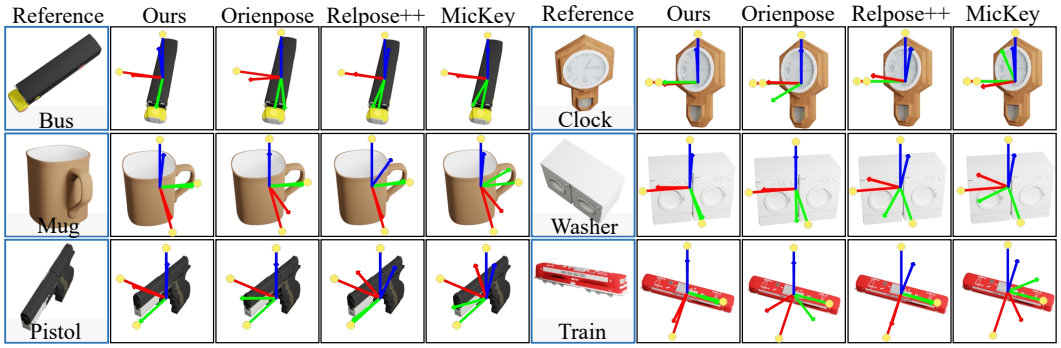

Figure 8: Additional qualitative results of the ShapeNet dataset. Lines with yellow points denote the GT axes.

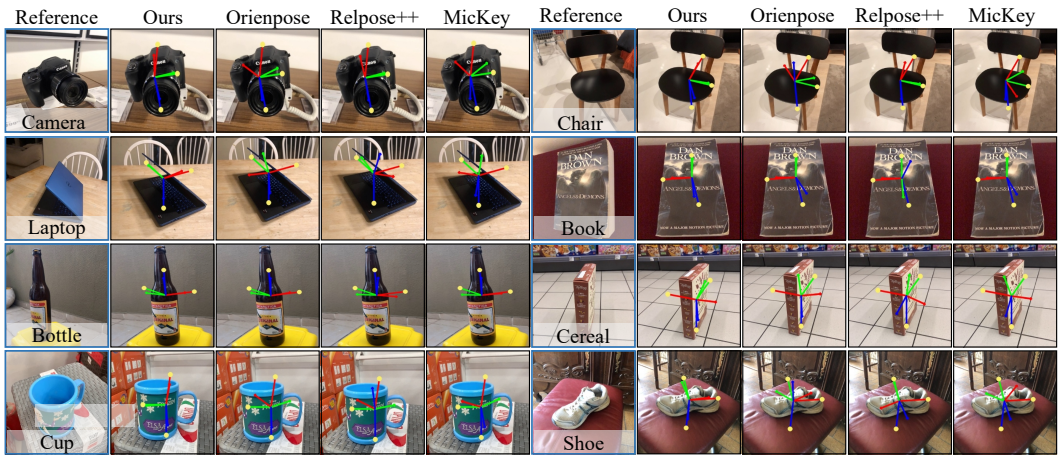

Figure 9: Additional qualitative results of the Objectron dataset. Lines with yellow points denote the GT axes.

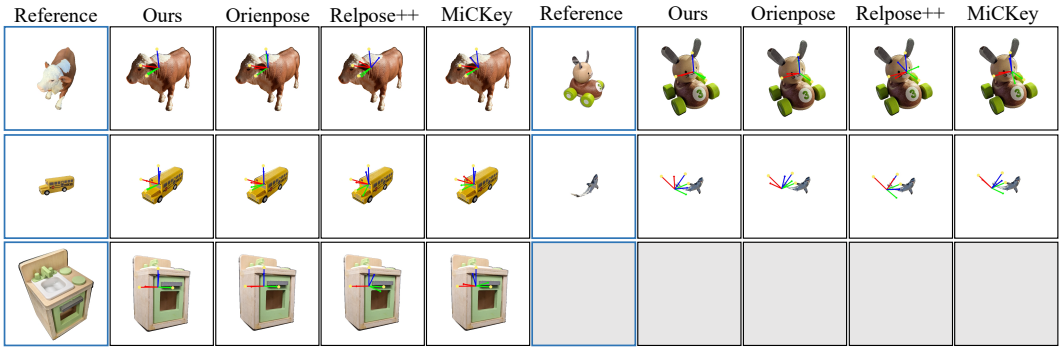

Figure 10: Additional qualitative results of the NAVI dataset. Lines with yellow points denote the GT axes.

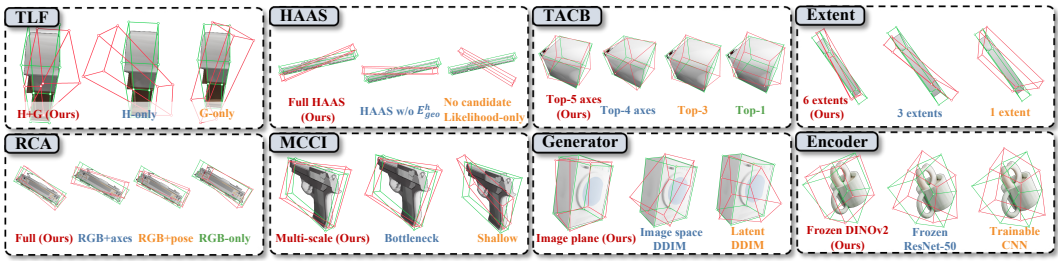

Figure 11: Qualitative results of the component-wise ablations. Green and red denote the ground-truth and predicted 3D bounding boxes, respectively. The comparisons cover TLF, HAAS, TACB, extent sampling, reference-coordinate anchoring, multi-scale conditional injection, axis generation, and reference encoding.

Inference Details. For each predicted axis likelihood field, we collect the 128 strongest local responses outside an 8-pixel neighbourhood of the predicted origin. A direction candidate is scored by combining its ray response, peak response, and agreement with the global direction using weights 0.6, 0.4, and 0.5, respectively. Directions separated by less than 5° are suppressed, after which the five highest-scoring candidates are retained for each axis. Their Cartesian product gives 125 target-view tri-axis hypotheses. TACB evaluates each hypothesis using projected axis extents in $\{80, 100, 120, 140, 160, 180\}$ pixels and two algebraic solution branches, resulting in at most 1,500 C2 rotation candidates. HAAS assigns weights of 1.0, 0.05, and 1.0 to directional reprojection error, algebraic instability, and TLF support, respectively. No iterative pose refinement or target ground-truth information is used during candidate selection. For visualization, all methods use the ground-truth translation and object dimensions, and therefore only the predicted rotation is varied. Further details of TACB and HAAS are provided in Appendices A.2 and A.3.

Robustness protocol. As shown in Figure 5, we evaluate robustness on the same ShapeNet test pairs used in the main evaluation and corrupt only the target RGB image, while keeping the reference image and geometric annotations unchanged. We consider severity levels $L \in \{10, 20, 30, 40\}$. Gaussian blur uses resolution-dependent odd kernel sizes controlled by factors $\{0.02, 0.04, 0.06, 0.08\}$, respectively. For occlusion, a square patch filled with uniform RGB noise is randomly placed within the object bounding box, with area ratios $\{0.05, 0.15, 0.25, 0.35\}$. The combined setting applies blur followed by occlusion at the corresponding level. All corruptions are generated deterministically with seed 42, and all conditions share the same test pairs and model checkpoint.

A.6 Additional Qualitative Results

Figures 8, 9, and 10 provide additional qualitative comparisons on ShapeNet, Objectron, and NAVI. Each example presents the reference image followed by the target-view predictions of AxisPose++ and the competing methods, where the red, green, and blue lines represent the predicted x, y, and z axes, respectively, and the lines with yellow points denote the GT axes. On the unseen ShapeNet categories, AxisPose++ produces coherent tri-axis configurations across objects with substantially different geometry and symmetry. The Objectron and NAVI examples further demonstrate that the learned reference-anchored axis transport generalizes to real images containing cluttered backgrounds, illumination changes, occlusion, and large appearance variations. In comparison, competing methods more frequently produce inconsistent axis directions or degenerate tri-axis configurations. These results complement the quantitative evaluation and illustrate the cross-category and synthetic-to-real generalization of AxisPose++.

A.7 Additional Ablation Results

Figure 11 provides qualitative examples for the component-wise ablations reported in the main paper. The ground-truth and predicted 3D bounding boxes are shown in green and red, respectively. The complete model generally produces better-aligned boxes than its ablated variants. The TLF comparison illustrates the complementary roles of dense likelihood evidence and global geome-

try, while the HAAS examples show the importance of jointly considering the available geometric and likelihood cues during hypothesis selection. The TACB and extent comparisons demonstrate that an adequate candidate search space is necessary to recover geometrically plausible rotations. The remaining examples examine reference-coordinate anchoring, multi-scale conditional injection, axis-generation paradigms, and reference encoders, further supporting the design choices adopted by AxisPose++.

Figure 12 further examines the influence of the TACB candidate budget on accuracy and inference time. Increasing the budget from Top-1 to Top-3 substantially improves Acc@30 from 51.7% to 63.4%, with inference time increasing from 0.168 to 0.295 seconds per frame. Top-5 reaches 68.2% at 0.364 seconds per frame. Further increasing the budget to Top-7 yields only a 0.62 percentage-point improvement, reaching 68.82%, while increasing inference time to 0.769 seconds per frame. We therefore adopt Top-5 as the default setting because it provides the best balance between pose accuracy and computational cost.

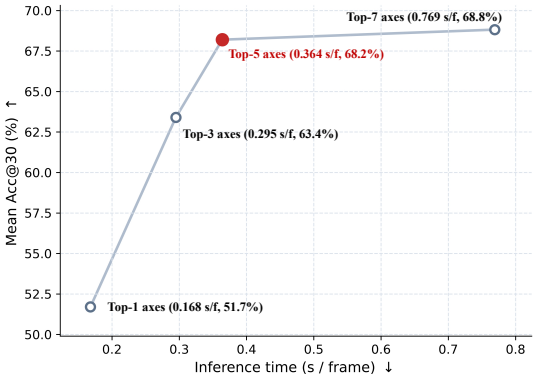

Figure 12: Accuracy-efficiency trade-off under different per-axis TACB candidate budgets. Top-5 achieves a favorable balance between Acc@30 and inference time, whereas Top-7 provides only a marginal accuracy gain at substantially higher computational cost.