MODEL-FREE 6D POSE ESTIMATION VIA 3D REGISTRATION WITH DISTANCE-COLOR FIELDS

Anonymous authors
Paper under double-blind review

ABSTRACT

We address model-free 6D pose estimation, which recovers the relative pose, *i.e.*, 3D translation and orientation, of an arbitrary object from a pair of reference and query RGB-D images, without requiring a pre-built CAD model. Unlike existing approaches, we tackle this problem directly in 3D space through two steps: (i) reconstructing a 3D mesh of the reference object using a generative reconstruction framework, and (ii) registering the query instance to the reconstructed mesh via 3D registration based on distance–color fields. Our amodal generator effectively reconstructs a complete 3D mesh aligned with the occluded reference view. Our Distance-Color Field Sample Consensus (DCF-SAC) efficiently performs 3D registration using distance-color fields of the reference mesh as fast lookup tables for sample-consensus scoring. On six benchmarks, the proposed method matches or outperforms reconstruction-based and reconstruction-free methods while running about four times faster than the leading reconstruction-based method.

1 INTRODUCTION

Object 6D pose estimation, recovering the rotation and translation of a rigid object from an observation, underlies robotic manipulation, augmented reality, and scene understanding. Most progress has relied on strong assumptions about the object. Instance-level methods train a network per object (Xiang et al., 2018; Peng et al., 2019), category-level methods rely on a canonical frame shared within a category (Wang et al., 2019), and model-based methods still require a textured CAD model at test time (Labbé et al., 2022; Nguyen et al., 2024; Wen et al., 2024). Model-free approaches eliminate the need for an explicit object model, but typically require numerous reference views or a video sequence of the object (He et al., 2022; Liu et al., 2022; Sun et al., 2022), which may not be practical in real-world setups. Single-reference pose estimation (Corsetti et al., 2024; Liu et al., 2025; Lee et al., 2025; Zuo et al., 2026) pushes this to the limit: a view with an object mask is the only prior, and the object may reappear in the query from any viewpoint and in any scene.

The main challenge of single-reference pose estimation is that the query view may depict an unseen side of the target object. Reconstruction-based methods (Lee et al., 2025; Wen et al., 2024) complete the unseen geometry with a 3D generative model, whose shape priors also help under occlusion, and then estimate the pose with an off-the-shelf render-and-compare estimator (Wen et al., 2024). However, rendering the mesh at every candidate pose dominates the runtime, and aligning the generated mesh to the reference view adds an error that carries into the query pose. Reconstruction-free methods (Corsetti et al., 2024; Liu et al., 2025; Zuo et al., 2026) regress the relative pose directly from the two views, bypassing single-view reconstruction entirely. However, the absence of an explicit 3D representation precludes geometric verification of pose hypotheses, leaving the estimation with no mechanism to detect or correct erroneous predictions.

In this paper, we address model-free 6D pose estimation directly in 3D space by connecting effective generative reconstruction to efficient 3D registration, bypassing render-and-compare pose estimation entirely (Fig. 1). This eliminates both the per-candidate rendering and the separate alignment of the mesh to the reference view, while preserving what makes reconstruction-based approaches attractive: a complete mesh that covers regions never visible in the reference view and explicit geometry against which pose hypotheses can be verified. The proposed method consists of two stages: (i) reference object reconstruction and (ii) registration with the distance-color fields. In the reconstruction stage, we reconstruct a complete 3D mesh aligned with the occluded reference view via amodal

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.

generative reconstruction. Our generator builds on (Park et al., 2026) and is further fine-tuned for amodal reconstruction (Wu et al., 2025), enabling it to complete a partially occluded reference into a full mesh. In the registration stage, we draw pose hypotheses from correspondences between the reference mesh and the query view and efficiently evaluate each hypothesis via Distance-Color Field Sample Consensus (DCF-SAC). The key idea is to convert the reference mesh's geometry and color into distance and color fields and use them as fast lookup tables for sample-consensus scoring of each hypothesis. On six benchmarks, our method matches or exceeds prior single-reference methods in accuracy while running about four times faster than the strongest reconstruction-based baseline, and ablations confirm that each component contributes.

Our contributions can be summarized as follows.

- We propose an efficient model-free 6D pose estimation pipeline that connects generative reconstruction directly to 3D registration with distance and color fields, removing the render-and-compare estimator and its reference alignment step.
- We improve generative reconstruction for this setting with amodal-aware fine-tuning and canonical-pose conditioning, yielding a complete mesh with a fixed reference pose even under partial occlusion.
- We propose DCF-SAC, a registration on distance-color fields in which one truncated cost both evaluates hypotheses densely over the query points and drives their lookup-based refinement.

2 Related Work

2.1 Model-free 6D Pose Estimation

Object 6D pose estimation classically matches or renders a known 3D model against the observation. Such models are rarely available in the wild, and model-free methods therefore replace them with reference observations in one of two schemes. The first builds a model: BundleSDF (Wen et al., 2023) and FoundationPose (Wen et al., 2024) fuse a posed RGB-D sequence into a neural object field, and Any6D (Lee et al., 2025) reconstructs a mesh from a single view and aligns it to that view with an off-the-shelf estimator. The second works without one: Oryon (Corsetti et al., 2024) registers vision-language features between the views, and One2Any (Liu et al., 2025) and CoordAR (Zuo et al., 2026) predict a coordinate map of the query relative to the reference. We follow the first scheme, but remove its reference pose estimate by generating the mesh at a canonical pose and absorb its reconstruction error by verifying the query pose over all query points.

2.2 Correspondence-based 3D Registration

Correspondence-based registration matches local descriptors, hand-crafted (Rusu et al., 2009) or learned (Choy et al., 2019; Poiesi & Boscaini, 2023; Hamza et al., 2025; Yu et al., 2021; Qin et al., 2022), solves pose hypotheses from the matches with RANSAC (Fischler & Bolles, 1981), and refines the selected one with ICP (Besl & McKay, 1992). This pipeline still dominates (Zhang et al., 2026), improved by outlier filtering (Bai et al., 2021; Zhang et al., 2023) and consensus sampling (Yan et al., 2025), while generative methods predict the transformation directly (Jiang et al., 2023; Pan et al., 2025). RANSAC variants sample minimal sets by pairwise distance consistency (Aiger et al., 2008; Yang et al., 2021; Chen et al., 2022; Zhang et al., 2023), soften the hard inlier count (Torr & Zisserman, 2000; Barath et al., 2020), or refit promising hypotheses (Chum et al., 2003). All of them score hypotheses on the correspondences, which are few and unreliable for a reconstructed mesh. We sample by the same distance consistency but evaluate every hypothesis by a cost over all query points.

2.3 Distance Fields for Pose Verification and Refinement

Geometric refinement and dense verification both pay a neighbor query per point. Distance fields answer such queries by lookup. Truncated signed distance fields are classically fused from range images (Curless & Levoy, 1996; Newcombe et al., 2011), and mesh-derived fields are evaluated on a regular grid by GPU closest-point queries (Macklin, 2022). A field also supplies normals from

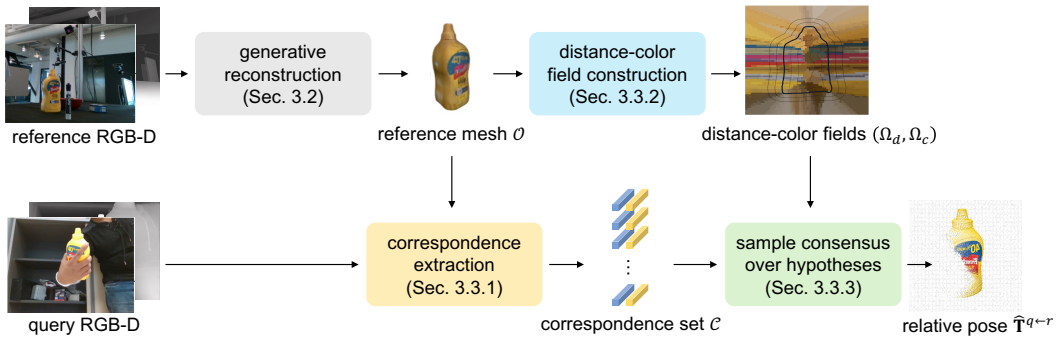

Figure 1: **Overall pipeline.** Generative reconstruction of reference (Sec. 3.2) reconstructs the reference object into a complete mesh $\mathcal{O}$ with a known pose $\mathbf{T}^{\mathrm{r}}$. 3D registration with distance-color field (Sec. 3.3) then registers $\mathcal{O}$ to the query view by sample consensus over hypotheses drawn from the correspondence set $\mathcal{C}$ and evaluated with the fields Ω_d and Ω_c, yielding $\hat{\mathbf{T}}^{\mathrm{q}\leftarrow\mathrm{r}}$.

its gradient, which SDF tracking exploits to refine a pose without neighbor search (Bylow et al., 2013). Photometric cues enter through color gradients in colored ICP (Park et al., 2017) or through rendering in iNeRF (Lin et al., 2021) and FoundationPose (Wen et al., 2024). All of them refine a single initialized pose and treat color separately. We construct distance and color fields on one lattice, whose lookups evaluate every hypothesis and refine the promising ones.

3 Proposed Method

3.1 Overview

Problem definition. Following prior work (Corsetti et al., 2024; Liu et al., 2025; Lee et al., 2025; Zuo et al., 2026), we formulate model-free 6D pose estimation as relative object pose estimation between two views of a novel object, a reference view and a query view, without any 3D model. Each view consists of color image $\mathbf{C} \in [0,1]^{h\times w\times 3}$, depth map $\mathbf{D} \in \mathbb{R}^{h\times w}$, object mask $\mathbf{M} \in \{0,1\}^{h\times w}$, and camera intrinsics $\mathbf{K} \in \mathbb{R}^{3\times 3}$. Given the reference view $(\mathbf{C}^{\mathrm{r}}, \mathbf{D}^{\mathrm{r}}, \mathbf{M}^{\mathrm{r}}, \mathbf{K}^{\mathrm{r}})$ and the query view $(\mathbf{C}^{\mathrm{q}}, \mathbf{D}^{\mathrm{q}}, \mathbf{M}^{\mathrm{q}}, \mathbf{K}^{\mathrm{q}})$, the goal is to estimate the relative pose $\mathbf{T}^{\mathrm{q}\leftarrow\mathrm{r}} = (\mathbf{R}, \mathbf{t}) \in \mathrm{SE}(3)$ that takes the object from the reference camera to the query camera, from which the query pose can be determined once the reference pose is known. For evaluation, the ground-truth poses of the object, $\mathbf{T}^{\mathrm{r}}_{\mathrm{gt}}$ and $\mathbf{T}^{\mathrm{q}}_{\mathrm{gt}}$, are given in the coordinate system of the object CAD model such that $\mathbf{T}^{\mathrm{q}\leftarrow\mathrm{r}}\,\mathbf{T}^{\mathrm{r}}_{\mathrm{gt}} = \mathbf{T}^{\mathrm{q}}_{\mathrm{gt}}$.

Overall pipeline. To estimate the relative pose $\mathbf{T}^{\mathrm{q}\leftarrow\mathrm{r}}$, we connect generative reconstruction of the reference object directly to 3D registration with distance-color fields in two stages. The first stage, *generative reconstruction of reference* (Sec. 3.2), reconstructs the reference object from the reference view into a complete mesh $\mathcal{O}$ in a frame of our choice, whose pose $\mathbf{T}^{\mathrm{r}}$ in the reference camera is therefore known without estimation. The second stage, *3D registration with distance-color field* (Sec. 3.3), registers $\mathcal{O}$ to the query view in three steps. *Correspondence extraction* (Sec. 3.3.1) matches descriptors between $\mathcal{O}$ and the query point cloud into the correspondence set $\mathcal{C}$, and *distance-color field construction* (Sec. 3.3.2) builds the distance field Ω_d and the color field Ω_c once from $\mathcal{O}$. *Sample consensus over hypotheses* (Sec. 3.3.3) then draws pose hypotheses from $\mathcal{C}$, evaluates and refines each by lookups into Ω_d and Ω_c, and rebases the best one through $\mathbf{T}^{\mathrm{r}}$ into the estimate $\hat{\mathbf{T}}^{\mathrm{q}\leftarrow\mathrm{r}}$. The overall pipeline is illustrated in Fig. 1.

3.2 Generative Reconstruction of Reference

In this stage, the reference object is reconstructed to a complete mesh in a generative manner, which serves as a stand-in for the missing full 3D model in the model-free 6D pose estimation. We denote this mesh by $\mathcal{O}$ and describe the architecture, training, and inference of our reconstructor below.

Architecture. Our reconstructor is a generative RGBD-to-mesh model built on the image-to-3D generation model, TRELLIS (Xiang et al., 2025), and its depth-aware variant introduced in Affos-

truction (Park et al., 2026). TRELLIS generates an object as the active voxels of a 64^3 grid, each carrying a local latent, in two steps: a flow transformer generates the voxel structure through a sparse structure VAE, and a sparse flow transformer generates the latents, which a decoder turns into a textured mesh. Both are trained with the conditional flow matching objective

$$\mathcal{L}_{\mathrm{CFM}} = \mathbb{E}_{t,\mathbf{z}_0,\boldsymbol{\epsilon}} \big\| \mathbf{v}(\mathbf{z}_t, \mathcal{V}, t) - (\boldsymbol{\epsilon} - \mathbf{z}_0) \big\|_2^2, \qquad \mathbf{z}_t = (1-t)\,\mathbf{z}_0 + t\,\boldsymbol{\epsilon}, \tag{1}$$

where $\mathbf{v}$ is the velocity predicted from the noisy latent $\mathbf{z}_t$ at time t and the conditioning tokens $\mathcal{V}$ attended by cross-attention, $\mathbf{z}_0$ is a clean latent, and $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. In TRELLIS, $\mathcal{V}$ are the DINOv2 (Oquab et al., 2023) features of an RGB image without depth information, and the object mesh is generated in its own canonical frame, unrelated to the observation in pose and scale. In contrast, Affostruction takes depth, intrinsics, and camera pose of the view as additional input and lifts the features $\mathcal{V}$ by aggregating them within each voxel of a coarse grid:

$$\mathcal{V} = \big\{ \bar{\mathbf{g}}_\ell + \mathrm{PE}(\mathbf{p}_\ell) \big\}_\ell, \tag{2}$$

where $\bar{\mathbf{g}}_\ell$ is the aggregated feature, and PE denotes a 3D positional encoding of the voxel position $\mathbf{p}_\ell$. Even from a single view, the visible surface thus enters at its observed location in a frame we choose, and the generated object shares this frame up to a known scale. We keep the sparse flow transformer and the decoder unchanged and retrain only the flow transformer, as described below.

Rotation-robust pretraining. Since the dataset used for training (Park et al., 2026; Fu et al., 2021; Collins et al., 2022; Khanna et al., 2024) stores 3D objects in its own canonical, upright frames, a generator trained on them learns to complete objects only in that orientation. This is problematic in our generic setup where a novel object can appear in the reference view in any orientation. We address this by rotating the conditioning voxel coordinates and the 64^3 target occupancy jointly by the relative rotation between the two render cameras of the object, sampled independently of the conditioning views. This rotation is replaced by the identity with a probability of 0.25 and is perturbed by a random rotation of up to 15°. As the pretrained SS-VAE reconstructs rotated grids poorly, we finetune it under the same augmentation to encode the rotated targets and train the flow transformer from scratch in its new latent space.

Amodal finetuning. The pretrained reconstructor is trained on unoccluded views, whereas the reference object is often partially occluded, and occluded pixels yield no voxels that a model trained on clean inputs can differentiate from empty space. Following Amodal3R (Wu et al., 2025), we finetune the flow transformer on synthetically occluded views and extend its cross-attention with two components adapted to voxel tokens. Occluders are cut from the silhouette and depth relief of another training object and mask both the color and the depth of the target, while the supervision remains over the complete structure. First, each visible token is weighted by its visibility $\omega_\ell \in (0, 1]$, the unoccluded fraction of target pixels in its projection averaged over the views, which adds $\log \omega_\ell$ to its attention logits. Second, the occluder relief is back-projected into the same grid, and its voxels form appearance-free tokens that carry only their 3D positional encoding and are attended by an additional cross-attention after the one over $\mathcal{V}$ in every block, which lets the model attribute missing tokens behind an occluder to occlusion rather than to free space. The new layers are randomly initialized, rotation augmentation remains active, and the condition is dropped for classifier-free guidance by emptying both token sets.

Pose conditioning at inference. At inference, we need the reference mesh together with its pose $\mathbf{T}^{\mathrm{r}}$ in the reference camera, which is required later to rebase the relative pose estimate (Sec. 3.3.3). Our reconstructor, built on Affostruction, takes the camera pose of the view as input and generates the mesh in the frame this pose defines. Specifically, the observed points of each view are transformed into the generator's unit cube by that view's camera pose, then voxelized and assigned their image features to form the tokens $\mathcal{V}$ in Eq. (2). Since the reference view has no camera pose of its own, we use the camera-aligned canonical pose as:

$$\mathbf{T}^{\mathrm{r}} = (\mathbf{R}_0, \mathbf{o}), \tag{3}$$

where $\mathbf{R}_0$ is the fixed axis flip between the camera and grid conventions. The translation $\mathbf{o}$ centers the observed points in-plane. In depth, it places them halfway between centering them in the cube and aligning their front surface with the front face of the cube, which leaves room for the unobserved back of the object. We then preprocess the depth map $\mathbf{D}^{\mathrm{r}}$ within the object mask $\mathbf{M}^{\mathrm{r}}$ following

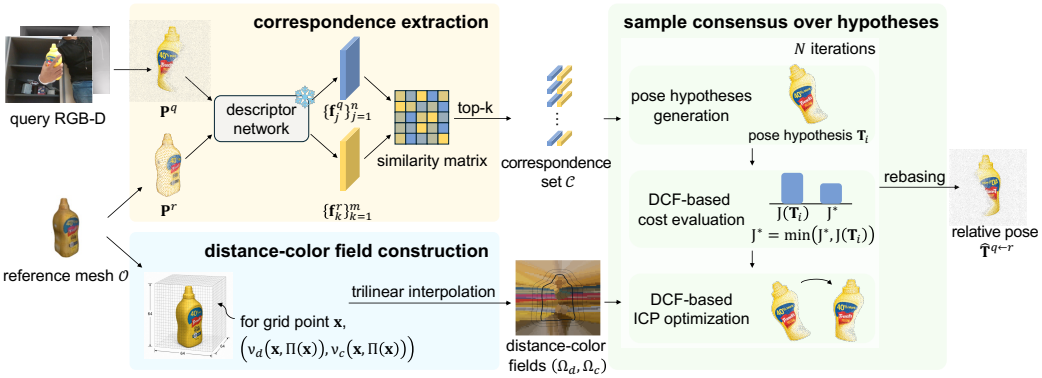

Figure 2: **3D Registration via Distance-Color Field SAmple Consensus.** Correspondence extraction (top) matches the query and reference point clouds into $\mathcal{C}$, and field construction (bottom) builds Ω_d and Ω_c once from $\mathcal{O}$. Sample consensus over hypotheses (right) draws pose hypotheses from $\mathcal{C}$, evaluates and refines them with the fields, and returns $\hat{\mathbf{T}}^{\mathrm{q}\leftarrow\mathrm{r}}$. See text for the details.

CoordAR (Zuo et al., 2026), rejecting depth values outside a range around the object center and refilling them from their nearest valid neighbors, with the range and the image crop set by an object diameter estimated from the mask bounding box and the median depth. The resulting points $\mathbf{p}$ in the reference camera then map into the cube as

$$\mathbf{p} \mapsto s\,\mathbf{R}_0^\top(\mathbf{p} - \mathbf{o}), \tag{4}$$

where s scales the larger in-plane extent of the points to the cube with a margin λ. As no occluder is known, we disable the occluder branch, and the finetuned model still completes occluded regions. Rescaling the generated mesh by $1/s$ gives $\mathcal{O} = (\mathbf{V}, \mathbf{F})$ at metric scale with vertices $\mathbf{V}$ and triangular faces $\mathbf{F}$, and each face f takes the mean color $\mathbf{c}_f \in [0,1]^3$ of its three vertices.

3.3 3D Registration with Distance-Color Field

In this stage, the reconstructed 3D mesh $\mathcal{O}$ is aligned to the query observation via efficient 3D registration. We represent the query view by the colored point cloud $\mathbf{P}^{\mathrm{q}} = \{(\mathbf{p}_j^{\mathrm{q}}, \mathbf{c}_j^{\mathrm{q}})\}_{j=1}^n$, obtained from the depth $\mathbf{D}^{\mathrm{q}}$ within $\mathbf{M}^{\mathrm{q}}$ using $\mathbf{K}^{\mathrm{q}}$ after discarding outliers, where each point $\mathbf{p}_j^{\mathrm{q}}$ takes the color $\mathbf{c}_j^{\mathrm{q}} \in [0,1]^3$ of its pixel in $\mathbf{C}^{\mathrm{q}}$. The reference view is represented by the colored reference point cloud $\mathbf{P}^{\mathrm{r}} = \{(\mathbf{p}_k^{\mathrm{r}}, \mathbf{c}_k^{\mathrm{r}})\}_{k=1}^m$ of m points sampled uniformly on $\mathcal{O}$, each taking the color of its supporting face. Each triple of descriptor correspondences between the two clouds fixes a pose hypothesis, but few hypotheses are close to the true pose. Verifying them by counting hard inliers as in RANSAC (Fischler & Bolles, 1981) is brittle under reconstruction error. We instead evaluate each hypothesis by a truncated cost over all query points, read from a distance field and a color field constructed once from $\mathcal{O}$, where one lookup gives each point its surface distance and color without nearest-neighbor search. We call this scheme Distance-Color Field SAmple Consensus (DCF-SAC), summarized in Alg. 1 (App. A.2), and detail its correspondence extraction (Sec. 3.3.1), field construction (Sec. 3.3.2), and sample consensus over hypotheses (Sec. 3.3.3) below.

3.3.1 Correspondence Extraction

We use a descriptor network $\Phi(\cdot)$ for both colored point clouds $\mathbf{P}^{\mathrm{r}}$ and $\mathbf{P}^{\mathrm{q}}$. The coordinates of each cloud are centered at their own centroid and divided by the diameter d of $\mathcal{O}$, while their colors remain unchanged, giving the normalized point clouds $\bar{\mathbf{P}}^{\mathrm{r}}$ and $\bar{\mathbf{P}}^{\mathrm{q}}$. Applying Φ to them yields per-point descriptors $\{\mathbf{f}_k^{\mathrm{r}}\}_{k=1}^m$ and $\{\mathbf{f}_j^{\mathrm{q}}\}_{j=1}^n$. In our implementation, we employ dGeDi (Hamza et al., 2025) as the feature extractor, which computes geometry-based descriptors without using the color information. Their similarities form the matrix $\mathbf{S} \in \mathbb{R}^{m\times n}$,

$$\mathbf{S}_{kj} = \frac{\langle \mathbf{f}_k^{\mathrm{r}}, \mathbf{f}_j^{\mathrm{q}} \rangle}{\|\mathbf{f}_k^{\mathrm{r}}\|\,\|\mathbf{f}_j^{\mathrm{q}}\|}. \tag{5}$$

Matching in both directions gives the correspondence set $\mathcal{C}$ of index pairs (k, j),

$$\mathcal{C} = \big\{(k, \arg\max_j \mathbf{S}_{kj})\big\}_{k=1}^m \cup \big\{(\arg\max_k \mathbf{S}_{kj}, j)\big\}_{j=1}^n. \tag{6}$$

The two terms give the best matching counterpart for each reference point and each query point.

3.3.2 Distance-Color Field Construction

Nearest-neighbor search requires a search per point and per hypothesis, and we therefore construct the geometry and the color of $\mathcal{O}$ once per object into a distance field and a color field over a regular lattice $\mathcal{X}$ of R^3 grid points $\mathbf{x} \in \mathbb{R}^3$ spanning the bounding box $\mathcal{B}$ of $\mathcal{O}$ padded by ηd on each side. Each $\mathbf{x}$ has an exact closest point $\Pi(\mathbf{x}) \in \mathcal{O}$ on the triangles $\mathbf{F}$, found by a GPU bounding-volume-hierarchy search (Macklin, 2022), and a constructor Γ turns a rule $\nu : \mathbb{R}^3 \times \mathbb{R}^3 \to \mathbb{R}^\kappa$ on $(\mathbf{x}, \Pi(\mathbf{x}))$ into a field evaluated at any $\mathbf{p} \in \mathcal{B}$ by trilinear interpolation,

$$\Gamma(\nu)(\mathbf{p}) = \sum_{\mathbf{x} \in \mathcal{N}(\mathbf{p})} w_{\mathbf{x}}(\mathbf{p})\, \nu\big(\mathbf{x}, \Pi(\mathbf{x})\big), \tag{7}$$

where $\mathcal{N}(\mathbf{p}) \subset \mathcal{X}$ are the eight grid points enclosing $\mathbf{p}$ and $w_{\mathbf{x}}(\mathbf{p})$ the trilinear weights. The distance field $\Omega_d = \Gamma(\nu_d)$ stores the unsigned distance to the mesh surface,

$$\nu_d\big(\mathbf{x}, \Pi(\mathbf{x})\big) = \|\mathbf{x} - \Pi(\mathbf{x})\|. \tag{8}$$

An inside/outside test requires $\mathcal{O}$ to be watertight, which a single-view reconstruction is not, and the cost never uses it: only the distance to the surface enters, and the field therefore carries no sign. The color field $\Omega_c = \Gamma(\nu_c)$ extends the surface color of $\mathcal{O}$ into the surrounding space,

$$\nu_c\big(\mathbf{x}, \Pi(\mathbf{x})\big) = \mathbf{c}_{f(\mathbf{x})}, \tag{9}$$

where $\mathbf{c}_{f(\mathbf{x})} \in [0,1]^3$ denotes the mean color of the face $f(\mathbf{x})$ of $\mathcal{O}$ containing $\Pi(\mathbf{x})$. Both fields depend only on $\mathcal{O}$. The color distance to a query point depends on the point's own color and is therefore not constructed but computed during cost evaluation (Sec. 3.3.3).

3.3.3 Sample Consensus over Hypotheses

Here, the correspondence set $\mathcal{C}$ (Sec. 3.3.1) provides minimal sets of three correspondences, each of which yields a pose hypothesis, and the fields Ω_d and Ω_c (Sec. 3.3.2) give every query point its distance to the surface and the surface color under that hypothesis at one lookup. We evaluate each hypothesis with these lookups, refine it if promising, and take the best one as the estimate $\hat{\mathbf{T}}^{\mathrm{q}\leftarrow\mathrm{r}}$.

Pose hypotheses generation. Each triple drawn from $\mathcal{C}$ is a minimal set, yielding a pose hypothesis $\mathbf{T}_i$ from the object frame to the query camera by Procrustes (Kabsch, 1976; Arun et al., 1987). Two correspondences (k, j) and (k', j') are congruent if

$$\big|\, \|\mathbf{p}^{\mathrm{r}}_k - \mathbf{p}^{\mathrm{r}}_{k'}\| - \|\mathbf{p}^{\mathrm{q}}_j - \mathbf{p}^{\mathrm{q}}_{j'}\| \,\big| < \varepsilon_c. \tag{10}$$

As random triples are mostly inconsistent, we draw them from a consistency graph connecting each congruent pair, built once before the loop: a random edge plus a third correspondence adjacent to both endpoints gives a congruent triple. At each iteration, one such triple is drawn and its pose hypothesis is immediately evaluated, with refinement performed only if admitted. We cap the loop at N iterations to prevent excessively long runs, although early stopping may terminate it sooner.

DCF-based cost evaluation. Each hypothesis $\mathbf{T}_i$ maps the query points into the object frame as $\mathbf{p}_{ij} = \mathbf{T}_i^{-1}\mathbf{p}^{\mathrm{q}}_j$, where one lookup per point reads both fields. The distance cost J_d penalizes the distance $\Omega_d(\mathbf{p}_{ij})$ of each point to the surface up to a truncation radius θ_d,

$$J_d(\mathbf{T}_i) = \frac{1}{n\,\theta_d^2} \sum_{j=1}^{n} \min\big(\Omega_d(\mathbf{p}_{ij})^2, \theta_d^2\big), \tag{11}$$

and the color cost J_c penalizes the difference between the surface color $\Omega_c(\mathbf{p}_{ij})$ read at each point and the point's own color $\mathbf{c}^{\mathrm{q}}_j$ up to a truncation radius θ_c,

$$J_c(\mathbf{T}_i) = \frac{1}{n\,\theta_c^2} \sum_{j=1}^{n} \min\big(\|\Omega_c(\mathbf{p}_{ij}) - \mathbf{c}^{\mathrm{q}}_j\|^2, \theta_c^2\big), \tag{12}$$

both normalized to $[0, 1]$. The final cost, which we minimize, weights the color cost by α,

$$J(\mathbf{T}_i) = J_d(\mathbf{T}_i) + \alpha\, J_c(\mathbf{T}_i). \tag{13}$$

Beyond its truncation radius, a point adds only a constant. The cost thus counts agreeing points rather than measuring how badly the others disagree, and the radii replace both an inlier gate and a penalty for uncovered points. All $N \cdot n$ evaluations are field lookups.

DCF-based ICP optimization. Since the admitted hypothesis still carries the noise from its minimal correspondences, we refine it by point-to-field ICP on the same cost J. Each iteration reads the distance and its gradient from Ω_d and the color and its Jacobian from Ω_c at the transformed points, linearizes Eq. (13) in a small rigid motion, and applies a damped Gauss–Newton step. Points beyond a truncation radius receive zero weight in this step, and refinement thus fits exactly the points that the cost counts as agreeing, without a separate inlier test. The cost is re-evaluated after every step, for at most L iterations, and the linearization is detailed in App. A.3.

Early stopping and final selection. Refining every hypothesis wastes computation on the many that are far from the true pose. The loop therefore tracks the lowest cost of an unrefined hypothesis J^*_{raw} and the lowest cost of any visited pose J^*, and admits a hypothesis to DCF-based ICP only if $J(\mathbf{T}_i) \le (1+\tau)\, J^*_{\mathrm{raw}}$, which admits those within a fraction τ of the best. Each admitted hypothesis undergoes up to L ICP iterations, with early termination if an iterate falls behind J^* or no longer lowers the cost. When the outer loop terminates, after at most N iterations, the pose that attained J^* is the estimated object pose in the query camera $\hat{\mathbf{T}}^{\mathrm{q}}$. Rebasing it through the reference pose $\mathbf{T}^{\mathrm{r}}$ gives the relative pose between the two views: $\hat{\mathbf{T}}^{\mathrm{q}\leftarrow\mathrm{r}} = \hat{\mathbf{T}}^{\mathrm{q}}(\mathbf{T}^{\mathrm{r}})^{-1}$.

4 Experiments

4.1 Experimental Settings

Datasets. We evaluate on six benchmarks: YCB-V (Xiang et al., 2018) with 21 objects in clutter and occlusion, LINEMOD (Hinterstoisser et al., 2012) with 15 texture-poor objects, HO3D (Hampali et al., 2020) and YCBInEOAT (Wen et al., 2020) with objects manipulated by a hand or a robot arm, REAL275 (Wang et al., 2019) with household objects from six categories, and Toyota-Light (Hodaň et al., 2018) with 21 objects under lighting changes. On the first four, the 4107, 3000, 2008, and 746 query views share 55, 15, 4, and 5 reference views, respectively, taken one per scene and object for YCB-V and LINEMOD and from Any6D (Lee et al., 2025) for HO3D and YCBInEOAT. REAL275 and Toyota-Light instead pair each of their 2000 query views with its own reference view from a different scene, following Oryon (Corsetti et al., 2024).

Evaluation. Following prior work, we report the area under the accuracy curve (AUC) of ADD and ADD-S on YCB-V, YCBInEOAT, and HO3D, ADD(S)-0.1d on LINEMOD, REAL275, and Toyota-Light, and the BOP average recall (AR) on all six. ADD is the mean distance between the model points under the estimated and the ground-truth pose, ADD-S uses the closest-point distance for symmetric objects, and the AUC is computed up to 10 cm (Xiang et al., 2018). ADD(S)-0.1d (Hinterstoisser et al., 2012) counts a pose as correct if its ADD, or ADD-S for symmetric objects, is below 10% of the object diameter d. AR (Hodaň et al., 2020) averages the recalls over thresholds of three errors: the visible surface discrepancy (VSD) between the visible object surfaces, the maximum symmetry-aware surface distance (MSSD) between model points in 3D, and its image-space counterpart, the maximum symmetry-aware projection distance (MSPD). We re-evaluate all baselines using their released code and checkpoints, with consistent data and protocols. As the symmetry annotations of the benchmarks come from independent sources and are inconsistent, we evaluate all methods with a revised annotation derived for every evaluation object by a single criterion (App. A.1), which also reports the results under the original annotation.

4.2 Implementation Details

Generative reconstruction. The flow transformer (12 blocks, width 768, 165M parameters) operates on a $16^3 \times 8$ latent with 16^3 conditioning voxels. After finetuning the sparse structure VAE

Table 1: Model-free 6D pose estimation results on six benchmarks and their mean (BOP AR). All baselines are reproduced and evaluated under the same protocol. Best in **bold**, second best in *italic*.

Method	YCB-V	YCBInEOAT	HO3D	LINEMOD	Real275	Toyota-Light	Mean
Oryon (Corsetti et al., 2024)	52.3	0.8	0.2	26.6	45.7	46.7	28.7
One2Any (Liu et al., 2025)	72.0	35.7	29.1	56.0	53.5	51.3	49.6
Any6D (Lee et al., 2025)	72.1	*64.1*	56.1	36.8	57.4	65.1	58.6
CoordAR (Zuo et al., 2026)	*81.8*	58.2	*61.1*	*74.5*	**79.3**	**77.2**	*72.0*
Ours	**84.9**	**69.2**	**66.0**	**77.0**	*78.3*	*74.2*	**74.9**

Table 2: Dataset-specific metrics: ADD(-S) AUC and ADD(S)-0.1d recall.

	YCB-V		YCBInEOAT		HO3D		LINEMOD	Real275	Toyota-Light
Method	ADD	ADD-S	ADD	ADD-S	ADD	ADD-S	ADD(S)-0.1d		
Oryon	57.8	82.9	0.0	0.5	0.0	0.3	21.1	39.5	57.9
One2Any	77.2	85.9	27.6	81.3	33.8	81.3	54.2	49.3	65.1
Any6D	65.3	86.8	*48.3*	*89.7*	54.4	*90.9*	32.3	59.2	66.3
CoordAR	**84.0**	*93.3*	**54.9**	83.4	**66.7**	88.7	*78.7*	*77.1*	**88.8**
Ours	*77.6*	**95.1**	38.6	**90.3**	*61.7*	**91.5**	**84.5**	**85.7**	*84.1*

Table 3: Runtime analysis (ms).

Method	Time
Oryon	$335.9_{\pm 9.4}$
One2Any	$\mathbf{180.9}_{\pm 5.8}$
Any6D	$1{,}811.9_{\pm 1.5}$
CoordAR	$367.2_{\pm 7.1}$
Ours	$\mathit{324.1}_{\pm 10.3}$

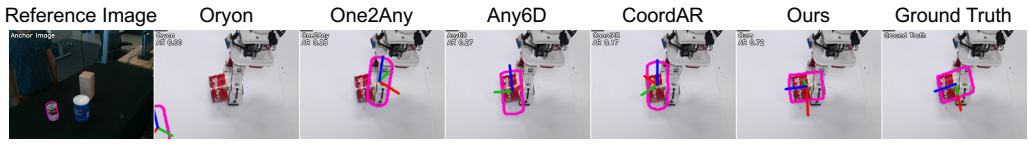

Figure 3: **Qualitative results.** Estimated pose (axes) and object outline (magenta) on the query.

for 50k steps, we pretrain it on the 155k training objects with the recipe of Affostruction (Park et al., 2026) (AdamW, learning rate 10^{-4}, batch size 64, EMA 0.9999, 1M steps on 8 GPUs). Amodal finetuning runs for 100k steps from the EMA weights (batch size 128, cosine schedule from 10^{-4}, one or two views) with occluders over 10–50% of the silhouette, rendered from a random view of another training object and randomly cropped, horizontally flipped with probability 0.5, rotated in-plane by up to $15°$, and resized, and the condition is dropped with probability 0.1 for classifier-free guidance. Inference uses 25 Euler steps, a guidance scale of 7, the margin $\lambda = 1/16$, and a depth-cleaning radius of 1.2 times half the estimated diameter.

DCF-SAC. For correspondence extraction, the query depth is eroded by 2 pixels and bilateral-filtered with radius 2, points beyond the mean plus 1.5 standard deviations of their distance to the centroid are removed, and multi-scale dGeDi (Hamza et al., 2025) yields 64-dimensional descriptors on $m = 2048$ points of $\mathcal{O}$ and 2048 query points, giving at most 4096 correspondences. Field construction uses $R = 64$ and $\eta = 0.25$ on one shared single-precision grid, with face colors averaged over vertices and ICP normals read from a grid of closest-point directions. We then draw $N = 20{,}000$ triples uniformly with $\varepsilon_c = 0.15\,d$ and solve each by the Kabsch algorithm (Kabsch, 1976), evaluate the cost with $\theta_d = 10\,\text{mm}$, $\theta_c = 0.4$ for RGB in $[0, 1]$, and $\alpha = 0.25$, run at most $L = 8$ ICP iterations with μ at 10^{-3} of the mean diagonal of the normal equations and color Jacobians by central differences over half a cell, and stop early with $\tau = 0.05$ for both admission and termination. Hypotheses are processed in GPU batches and replayed in drawing order.

4.3 Model-free 6D Pose Estimation

We compare our method with reconstruction-free and reconstruction-based baselines. Our method achieves the best mean AR, surpassing the reconstruction-based Any6D (Lee et al., 2025) on every benchmark and the strongest reconstruction-free method, CoordAR (Zuo et al., 2026), on four of the six (Tab. 1). The gains are largest under heavy occlusion by hands and grippers, where the reconstruction-free methods degrade most and the completed reference mesh keeps the unseen geometry available for registration, whereas CoordAR remains ahead on the two cross-scene benchmarks (Tab. 1). In the benchmark-specific metrics (Tabs. 2 and 2), our method leads in the

Table 4: Query pose estimators on our mesh.

Estimator	AR	Time (ms)
FoundationPose	74.90	2,170
DCF-SAC (ours)	74.92	267

Table 5: Amodal finetuning of the reconstructor.

Reconstructor	AR
Affostruction	74.75
+ amodal finetuning	74.92

Table 6: Correspondence selection.

Matching	$\lvert\mathcal{C}\rvert$	AR
Static top-k	2048	68.63
Row top-k	2048	74.81
Column top-k	2003	73.75
Mutual top-k	62	68.74
Union top-k (ours)	3989	74.92

Table 7: Per-point cost for evaluation and ICP.

Variant	Per-point cost	AR	Time(s)
Point-to-point	$\lVert x - \mathrm{NN}(x)\rVert$	71.08	6.037
Point-to-plane	$\langle x - \mathrm{NN}(x),\, n\rangle$	74.44	6.057
Point-to-face	$d(x, \triangle)$	75.06	100.379
Point-to-field (J_d)	$\min(s(x)^2, \theta_d^2)$	73.94	0.308
Point-to-field (J_c)	$\min(\lVert\Delta c\rVert^2, \theta_c^2)$	22.73	0.333
Point-to-field (ours)	$J_d + \alpha J_c$	74.92	0.324

Table 8: Truncation radii θ_d and θ_c.

θ_d (mm)	AR	θ_c (RGB)	AR
5	73.03	0.20	73.75
7	74.09	0.30	73.92
10	74.92	0.40	74.92
14	73.24	0.50	74.70
20	71.29	0.60	74.61

symmetry-aware ADD-S and in most ADD(S)-0.1d results but trails CoordAR in ADD, suggesting that the registration aligns the surface accurately but occasionally settles on a symmetric counterpart of the true pose. These gains come at a runtime comparable to the reconstruction-free methods and several times shorter than that of Any6D (Tab. 3). Qualitative results are shown in Fig. 3. Compared to CoordAR (Zuo et al., 2026), our method is more robust to occlusion of the object in the query view. Additional qualitative results are provided in the supplementary material (Apps. A.7).

4.4 Ablation Study

We ablate each design choice by changing one component at a time while keeping the rest of the pipeline fixed. Replacing DCF-SAC with FoundationPose (Wen et al., 2024) on the same reconstructed mesh separates the registration from the reconstruction (Tab. 4), and DCF-SAC reaches the same accuracy at a fraction of the runtime, showing that removing rendering costs no accuracy. As the reference object is often partially occluded, amodal finetuning improves the accuracy by completing the occluded parts of the reference mesh that the registration relies on (Tab. 5). Among the rules for selecting $\mathcal{C}$ from $\mathbf{S}$ (Tab. 6), the union of row and column selection retains the most matches and yields the best accuracy, whereas mutual selection keeps too few correspondences and static selection falls far behind. Among the per-point costs for evaluation and refinement (Tab. 7), the field-based costs avoid the nearest-neighbor search of conventional ICP distances, and combining the distance and color terms is best, as color separates poses that geometry alone cannot. For the truncation radii θ_d and θ_c in Eqs. (11) and (12), accuracy remains stable across the tested values (Tab. 8). Additional ablations on the query outlier removal, the distance field, the descriptor network, and the color field, as well as hyper-parameter studies, are provided in the supplementary material (Apps. A.5 and A.6).

5 Conclusion

We have presented a model-free 6D pose estimation method that connects generative reconstruction directly to 3D registration with distance-color fields. An amodal-aware generator produces a complete mesh aligned with the reference view by construction, and DCF-SAC evaluates and refines pose hypotheses by field lookups instead of rendering. On six benchmarks, our method matches or outperforms prior methods while running several times faster than render-and-compare estimation, and resolving the ambiguity of symmetric objects remains future work.

AI USE STATEMENT

In this work, we used generative AI tools to implement the methods that we drafted, clean and reformat datasets, and assist with translation. All outcomes were verified by the authors. We have not used generative AI tools to generate synthetic datasets, develop theoretical models or conceptual frameworks, formulate or prove mathematical claims, propose or refine hypotheses, design the methodology or experiments, analyze qualitative data, or interpret results, and the remaining tasks with required disclosure are not applicable to this work. Additionally, we used generative AI tools to refactor code, write batch-processing scripts, draft parts of the paper from human-written documents and prompts, improve the readability and grammar of the manuscript, and format references and tables. We have reviewed all AI-assisted work: the authors checked and tested the implementation, the data processing, and all scripts; all reported numbers come from our own experiments; and all drafted or revised text was checked against the method, the experimental results, and the cited works. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Dror Aiger, Niloy J. Mitra, and Daniel Cohen-Or. 4-points congruent sets for robust pairwise surface registration. *ACM Transactions on Graphics*, 27(3):85:1–85:10, 2008. doi: 10.1145/1360612.1360684.

Sheng Ao, Qingyong Hu, Hanyun Wang, Kai Xu, and Yulan Guo. BUFFER: Balancing Accuracy, Efficiency, and Generalizability in Point Cloud Registration. In *CVPR*, 2023.

K. S. Arun, Thomas S. Huang, and Steven D. Blostein. Least-squares fitting of two 3-d point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(5):698–700, 1987. doi: 10.1109/TPAMI.1987.4767965.

Xuyang Bai, Zixin Luo, Lei Zhou, Hongkai Chen, Lei Li, Zeyu Hu, Hongbo Fu, and Chiew-Lan Tai. PointDSC: Robust point cloud registration using deep spatial consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15859–15869, 2021. doi: 10.1109/CVPR46437.2021.01560.

Daniel Barath, Jana Noskova, Maksym Ivashechkin, and Jiri Matas. MAGSAC++, a fast, reliable and accurate robust estimator. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1304–1312, 2020. doi: 10.1109/CVPR42600.2020.00138.

Paul J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992. doi: 10.1109/34.121791.

Erik Bylow, Jürgen Sturm, Christian Kerl, Fredrik Kahl, and Daniel Cremers. Real-time camera tracking and 3d reconstruction using signed distance functions. In *Proceedings of Robotics: Science and Systems (RSS)*, 2013. doi: 10.15607/RSS.2013.IX.035.

Zhi Chen, Kun Sun, Fan Yang, and Wenbing Tao. SC2-PCR: A second order spatial compatibility for efficient and robust point cloud registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13221–13231, 2022. doi: 10.1109/CVPR52688.2022.01287.

Christopher Choy, Jaesik Park, and Vladlen Koltun. Fully convolutional geometric features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8958–8966, 2019. doi: 10.1109/ICCV.2019.00905.

Ondrej Chum, Jiri Matas, and Josef Kittler. Locally optimized ransac. In *Pattern Recognition: 25th DAGM Symposium*, pp. 236–243. Springer, 2003.

Jasmine Collins, Shubham Goel, Kenan Deng, Achleshwar Luthra, Leon Xu, Erhan Gundogdu, Xi Zhang, Tomas F. Yago Vicente, Thomas Dideriksen, Himanshu Arora, Matthieu Guillaumin, and Jitendra Malik. ABO: Dataset and benchmarks for real-world 3d object understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Jaime Corsetti, Davide Boscaini, Changjae Oh, Andrea Cavallaro, and Fabio Poiesi. Open-vocabulary object 6d pose estimation. In *IEEE/CVF Computer Vision and Pattern Recognition Conference (CVPR)*, 2024.

Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, pp. 303–312, 1996. doi: 10.1145/237170.237269.

Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24 (6):381–395, 1981. doi: 10.1145/358669.358692.

Huan Fu, Rongfei Jia, Lin Gao, Mingming Gong, Binqiang Zhao, Steve Maybank, and Dacheng Tao. 3D-FUTURE: 3d furniture shape with texture. *International Journal of Computer Vision (IJCV)*, 129:3313–3337, 2021.

Shreyas Hampali, Mahdi Rad, Markus Oberweger, and Vincent Lepetit. HOnnotate: A method for 3d annotation of hand and object poses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3196–3206, 2020.

Amir Hamza, Andrea Caraffa, Davide Boscaini, and Fabio Poiesi. Distilling 3D distinctive local descriptors for 6D pose estimation. In *IROS*, 2025.

Yisheng He, Yao Wang, Haoqiang Fan, Jian Sun, and Qifeng Chen. FS6D: Few-shot 6d pose estimation of novel objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6814–6824, 2022.

Stefan Hinterstoisser, Vincent Lepetit, Slobodan Ilic, Stefan Holzer, Gary Bradski, Kurt Konolige, and Nassir Navab. Model based training, detection and pose estimation of texture-less 3d objects in heavily cluttered scenes. In *Proceedings of the Asian Conference on Computer Vision (ACCV)*, pp. 548–562, 2012.

Tomáš Hodaň, Frank Michel, Eric Brachmann, Wadim Kehl, Anders Glent Buch, Dirk Kraft, Bertram Drost, Joel Vidal, Stephan Ihrke, Xenophon Zabulis, Caner Sahin, Fabian Manhardt, Federico Tombari, Tae-Kyun Kim, Jiří Matas, and Carsten Rother. BOP: Benchmark for 6d object pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 19–34, 2018.

Tomáš Hodaň, Martin Sundermeyer, Bertram Drost, Yann Labbé, Eric Brachmann, Frank Michel, Carsten Rother, and Jiří Matas. BOP challenge 2020 on 6d object localization. In *European Conference on Computer Vision Workshops (ECCVW)*, pp. 577–594, 2020.

Haobo Jiang, Mathieu Salzmann, Zheng Dang, Jin Xie, and Jian Yang. SE(3) diffusion model-based point cloud registration for robust 6d object pose estimation. In *Advances in Neural Information Processing Systems*, volume 36, pp. 21285–21297, 2023.

Wolfgang Kabsch. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A*, 32(5):922–923, 1976. doi: 10.1107/S0567739476001873.

Mukul Khanna, Yongsen Mao, Hanxiao Jiang, Sanjay Haresh, Brennan Shacklett, Dhruv Batra, Alexander Clegg, Eric Undersander, Angel X. Chang, and Manolis Savva. Habitat synthetic scenes dataset (HSSD-200): An analysis of 3d scene scale and realism tradeoffs for objectgoal navigation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Yann Labbé, Lucas Manuelli, Arsalan Mousavian, Stephen Tyree, Stan Birchfield, Jonathan Tremblay, Justin Carpentier, Mathieu Aubry, Dieter Fox, and Josef Sivic. MegaPose: 6d pose estimation of novel objects via render & compare. In *Conference on Robot Learning (CoRL)*, 2022.

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 Computer Vision and Pattern Recognition Conference (CVPR)*, 2025.

Yen-Chen Lin, Pete Florence, Jonathan T. Barron, Alberto Rodriguez, Phillip Isola, and Tsung-Yi Lin. iNeRF: Inverting neural radiance fields for pose estimation. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1323–1330, 2021. doi: 10.1109/IROS51168.2021.9636708.

Mengya Liu, Siyuan Li, Ajad Chhatkuli, Prune Truong, Luc Van Gool, and Federico Tombari. One2any: One-reference 6d pose estimation for any object. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 6457–6467, 2025.

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 *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 298–315, 2022.

Miles Macklin. Warp: A High-performance Python Framework for GPU Simulation and Graphics. `https://github.com/nvidia/warp`, March 2022. NVIDIA GPU Technology Conference (GTC).

Richard A. Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J. Davison, Pushmeet Kohli, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. KinectFusion: Real-time dense surface mapping and tracking. In *2011 10th IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 127–136, 2011. doi: 10.1109/ISMAR.2011.6092378.

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 (CVPR)*, pp. 9903–9913, 2024.

Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023.

Yue Pan, Tao Sun, Liyuan Zhu, Lucas Nunes, Iro Armeni, Jens Behley, and Cyrill Stachniss. Register any point: Scaling 3d point cloud registration by flow matching. *arXiv preprint*, 2025.

Chunghyun Park, Seunghyeon Lee, and Minsu Cho. Affostruction: 3d affordance grounding with generative reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Colored point cloud registration revisited. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 143–152, 2017. doi: 10.1109/ICCV.2017.25.

Sida Peng, Yuan Liu, Qixing Huang, Xiaowei Zhou, and Hujun Bao. 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.

Fabio Poiesi and Davide Boscaini. Learning general and distinctive 3d local deep descriptors for point cloud registration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3): 3979–3985, 2023. doi: 10.1109/TPAMI.2022.3175371.

Zheng Qin, Hao Yu, Changjian Wang, Yulan Guo, Yuxing Peng, and Kai Xu. Geometric transformer for fast and robust point cloud registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11143–11152, 2022. doi: 10.1109/CVPR52688.2022.01086.

Radu Bogdan Rusu, Nico Blodow, and Michael Beetz. Fast point feature histograms (FPFH) for 3d registration. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3212–3217, 2009. doi: 10.1109/ROBOT.2009.5152473.

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 (CVPR)*, pp. 6825–6834, 2022.

Philip H. S. Torr and Andrew Zisserman. MLESAC: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78(1):138–156, 2000. doi: 10.1006/cviu.1999.0832.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2642–2651, 2019.

Bowen Wen, Chaitanya Mitash, Baozhang Ren, and Kostas E. Bekris. se(3)-TrackNet: Data-driven 6d pose tracking by calibrating image residuals in synthetic domains. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 10367–10373, 2020.

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 *CVPR*, 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 (CVPR)*, pp. 17868–17879, 2024. doi: 10.1109/CVPR52733.2024.01692.

Junting Wu, Chuanxia Zheng, Andrea Vedaldi, et al. Amodal 3d reconstruction from occluded 2d images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025.

Jianfeng Xiang et al. Structured 3d latents for scalable and versatile 3d generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR)*, 2025.

Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. PoseCNN: A convolutional neural network for 6d object pose estimation in cluttered scenes. In *Robotics: Science and Systems (RSS)*, 2018.

Shaocheng Yan, Yiming Wang, Kaiyan Zhao, Pengcheng Shi, Zhenjun Zhao, Yongjun Zhang, and Jiayuan Li. HeMoRa: Unsupervised heuristic consensus sampling for robust point cloud registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1363–1373, 2025. doi: 10.1109/CVPR52734.2025.00135.

Heng Yang, Jingnan Shi, and Luca Carlone. TEASER: Fast and certifiable point cloud registration. *IEEE Transactions on Robotics*, 37(2):314–333, 2021. doi: 10.1109/TRO.2020.3033695.

Hao Yu, Fu Li, Mahdi Saleh, Benjamin Busam, and Slobodan Ilic. CoFiNet: Reliable coarse-to-fine correspondences for robust point cloud registration. In *Advances in Neural Information Processing Systems*, volume 34, pp. 23872–23884, 2021.

Xiyu Zhang, Jiaqi Yang, Shikun Zhang, and Yanning Zhang. 3d registration with maximal cliques. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17745–17754, 2023. doi: 10.1109/CVPR52729.2023.01702.

Yu-Xin Zhang, Jie Gui, Baosheng Yu, Xiaofeng Cong, Xin Gong, Wenbing Tao, and Dacheng Tao. Deep learning-based point cloud registration: A comprehensive survey and taxonomy. *International Journal of Computer Vision*, 134(4), 2026. doi: 10.1007/s11263-025-02723-w. Article 168.

Dexin Zuo, Ang Li, Wei Wang, Wenxian Yu, and Danping Zou. Coordar: One-reference 6d pose estimation of novel objects via autoregressive coordinate map generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 14122–14130, 2026.

A Appendix

A.1 Symmetry Annotation

Symmetry enters the evaluation in two ways: MSSD and MSPD take the minimum error over the symmetry transformations of an object, and ADD(S)-0.1d applies ADD-S to symmetric objects. To evaluate all six datasets under one convention, we measure the symmetry of every evaluation object with a single criterion and apply the result to all methods.

Motivation. The symmetries used in prior evaluations come from four sources, each built for its own purpose: the BOP (Hodaň et al., 2018) `models_info` files for YCB-V, LINEMOD, and Toyota-Light; the BOP list `symmetric_obj_ids` of the objects evaluated with ADD-S; the FoundationPose (Wen et al., 2024) overrides for YCB-V; and the `models_info` file of Oryon (Corsetti et al., 2024) for REAL275. HO3D and YCBInEOAT contain only YCB-V objects and reuse its BOP `models_info` and FoundationPose overrides, without a `symmetric_obj_ids` list. Built independently, these sources differ in coverage and in how they describe a symmetry group. The YCB-V foam brick, for example, is C_2 in BOP but D_2 in FoundationPose, and rotationally symmetric objects such as the YCB-V mustard bottle and the Toyota-Light cans have no entry. For REAL275, the continuous axes lie along the model z-axis while the evaluation meshes are y-up, and the two Canon cameras carry a continuous axis, presumably from name-based category selection.

Criterion. We adopt the acceptance rule of the BOP Challenge (Hodaň et al., 2020), under which a rotation $\mathbf{Q}$ is a symmetry of a model with vertices $\mathbf{V}$ if

$$h\big(\mathbf{V}, \mathbf{QV}\big) < \epsilon, \qquad \epsilon = \max(15\,\mathrm{mm},\, 0.1d), \tag{14}$$

where h is the Hausdorff distance and d the model diameter. Each candidate axis is fitted in direction and position by minimizing the mean surface displacement and then tested by Eq. (14). An n-fold symmetry, $n = 2, \ldots, 6$, is accepted if all rotations $2\pi k/n$ and all divisors of n pass, and an axis that passes every order is annotated as continuous, $\mathrm{SO}(2)$. Perpendicular two-fold axes through the principal axis extend C_n to the dihedral group D_n, which is accepted only if every element of the group satisfies Eq. (14). We do not reproduce the second step of BOP, which keeps only the symmetries that texture cannot resolve and relies on the judgment of the organizers. To stay close to the original annotation, we add the measured symmetries to the existing entries of YCB-V, LINEMOD, and Toyota-Light without removing any, and derive the REAL275 annotation anew, as its axes are defined in a frame different from that of the meshes.

Results. Tab. 9 compares the measured groups with the original ones. In total, 24 objects receive a continuous axis and 23 a discrete group (twelve C_2, seven D_2, three D_4, and one D_6). The measured groups agree with most declared ones, including the four boxes of FoundationPose, and add a symmetry to thirteen objects of YCB-V, LINEMOD, and Toyota-Light. In REAL275, the bottles, bowls, and cans keep a continuous axis, now along the y-axis of the meshes, the two cameras lose theirs, and the two laptops receive C_2. Without the texture-based step, a few accepted symmetries (†) may be resolvable by texture; they stem from the lower bound of ϵ or its growth with the diameter, and we keep them to avoid manual selection. All results in Tabs. 1–2 use this revised annotation, and Tabs. 10–12 report them under the original one for reference.

A.2 DCF-SAC Algorithm

Alg. 1 summarizes DCF-SAC (Sec. 3.3) as a loop over N pose hypotheses. Before the loop, the congruent pairs of $\mathcal{C}$ are connected into a consistency graph, from which each iteration draws a congruent triple and solves its pose $\mathbf{T}_i$ by Procrustes. The cost $J(\mathbf{T}_i)$ is read from Ω_d and Ω_c with one lookup per query point, and a hypothesis whose cost exceeds the lowest unrefined cost J^*_{raw} by more than a factor of $1+\tau$ is dropped without refinement. The remaining hypotheses are refined by at most L steps of DCF-based ICP, each kept only if it lowers the cost and stays within a factor of $1+\tau$ of the lowest cost J^*. The pose $\hat{\mathbf{T}}^{\mathrm{q}}$ that attains J^* is finally rebased through $\mathbf{T}^{\mathrm{r}}$ into $\hat{\mathbf{T}}^{\mathrm{q}\leftarrow\mathrm{r}}$.

Table 9: Revised symmetry annotation of benchmark datasets. *Shipped*: original group and its source (BOP, FoundationPose (FP), or Oryon), with — for no entry. h: maximum Hausdorff displacement over the accepted group, compared with ϵ in Eq. (14). (conf.): original group confirmed by measurement. †: possibly resolvable by texture.

Dataset	Object	Shipped	Ours	h (mm)	ϵ (mm)
YCB-V (HO3D, YCBInEOAT)	2 cracker box	D_2 (FP)	D_2 (conf.)	7.2	27.0
	3 sugar box	D_2 (FP)	D_2 (conf.)	5.2	19.8
	5 mustard bottle	—	C_2	3.6	19.6
	7 pudding box	—	D_4†	14.1	15.0
	8 gelatin box	—	D_4†	9.9	15.0
	9 potted meat can	D_2 (FP)	D_2 (conf.)	6.2	15.0
	12 bleach cleanser	—	C_2	16.2	26.0
	16 wood block	D_4 (BOP)	$\mathrm{SO}(2)$†	16.8	23.7
	17 scissors	—	C_2	12.5	20.4
	19 large clamp	C_2 (BOP)	C_2 (conf.)	6.6	17.5
	20 extra large clamp	C_2 (BOP)	C_2 (conf.)	6.9	21.7
	21 foam brick	D_2 (FP)	D_4	13.6	15.0
LINEMOD	10 eggbox	C_2 (BOP)	C_2 (conf.)	11.3	16.5
	11 glue	C_2 (BOP)	C_2 (conf.)	9.2	17.6
	12 holepuncher	—	C_2	13.3	15.0
Toyota-Light	1 remote	—	D_2	11.6	15.2
	2 magazine	—	D_2	12.0	21.9
	8 plastic bottle	—	$\mathrm{SO}(2)$	5.2	24.3
	9 croc	—	C_2†	26.5	30.4
	14 cracker box	—	D_2	4.4	27.5
	15 milk carton	C_2 (BOP)	D_6†	25.9	26.3
	16 plastic container	C_2 (BOP)	D_2	11.0	21.3
	17 can	—	$\mathrm{SO}(2)$	5.0	15.0
	18 can	—	$\mathrm{SO}(2)$	2.6	15.0
	19 basket	C_2 (BOP)	C_2 (conf.)	9.6	33.8
Real275	3 bottles, 3 bowls, 3 cans	$\mathrm{SO}(2)$, z (Oryon)	$\mathrm{SO}(2)$, fitted y	4.3–15.5	15.0–19.3
	camera canon len	$\mathrm{SO}(2)$ (Oryon)	none	27.0	18.2
	camera canon wo len	$\mathrm{SO}(2)$ (Oryon)	none	23.9	15.8
	laptop air xin	—	C_2†	39.7	40.0
	laptop alienware	—	C_2†	26.2	45.7

Table 10: BOP AR under the original symmetry annotation, as in Tab. 1.

Method	YCB-V	YCBInEOAT	HO3D	LINEMOD	Real275	Toyota-Light	Mean
Oryon (Corsetti et al., 2024)	51.6	0.6	0.2	26.1	43.2	41.3	27.2
One2Any (Liu et al., 2025)	72.0	35.7	29.1	55.9	53.2	48.1	49.0
Any6D (Lee et al., 2025)	71.4	*64.1*	56.1	36.5	53.6	60.2	57.0
CoordAR (Zuo et al., 2026)	*81.5*	58.2	*61.1*	*74.0*	**75.7**	**76.8**	*71.2*
Ours	**85.5**	**65.7**	**63.5**	**77.2**	*72.0*	*68.0*	**72.0**

Table 11: ADD/ADD-S AUC under the original annotation, as in Tab. 2.

	YCB-V		YCBInEOAT		HO3D	
Method	ADD	ADD-S	ADD	ADD-S	ADD	ADD-S
Oryon	57.9	82.9	0.0	0.5	0.0	0.3
One2Any	77.2	85.9	27.6	81.3	33.8	81.3
Any6D	65.3	86.8	*48.3*	**89.7**	54.4	*90.9*
CoordAR	**84.0**	*93.3*	**54.9**	83.4	**66.7**	88.7
Ours	*77.6*	**95.4**	38.6	**89.7**	*60.7*	**91.0**

Table 12: ADD(S)-0.1d recall under the original annotation, as in Tab. 2.

	LINEMOD	Real275	Toyota-Light
Method	ADD(S)-0.1d		
Oryon	16.4	33.5	32.0
One2Any	49.6	50.5	42.5
Any6D	29.7	57.0	52.0
CoordAR	*76.8*	**83.0**	**79.0**
Ours	**82.9**	*79.5*	*61.0*

Algorithm 1 Pose estimation with DCF-SAC (sample consensus with early stopping)

Require: colored reference and query point cloud $\mathbf{P}^{\mathrm{r}} = \{(\mathbf{p}_k^{\mathrm{r}}, \mathbf{c}_k^{\mathrm{r}})\}_{k=1}^{m}$, $\mathbf{P}^{\mathrm{q}} = \{(\mathbf{p}_j^{\mathrm{q}}, \mathbf{c}_j^{\mathrm{q}})\}_{j=1}^{n}$, correspondence set $\mathcal{C}$, distance field Ω_d and color field Ω_c constructed from $\mathcal{O}$, reference pose $\mathbf{T}^{\mathrm{r}}$, hyper-parameters $N, L, \varepsilon_c, \theta_d, \theta_c, \alpha, \mu, \tau$
Ensure: relative pose $\hat{\mathbf{T}}^{\mathrm{q}\leftarrow\mathrm{r}}$

Initialization
Build the consistency graph on $\mathcal{C}$, connecting each congruent pair $(k, j), (k', j')$ with $\big|\, \|\mathbf{p}_k^{\mathrm{r}} - \mathbf{p}_{k'}^{\mathrm{r}}\| - \|\mathbf{p}_j^{\mathrm{q}} - \mathbf{p}_{j'}^{\mathrm{q}}\| \,\big| < \varepsilon_c$
$J^* \leftarrow \infty, \quad J^*_{\mathrm{raw}} \leftarrow \infty$ ▷ lowest cost, and lowest of an *unrefined* one; $\hat{\mathbf{T}}^{\mathrm{q}}$ attains J^*
Sample consensus over hypotheses
for $i = 1, \ldots, N$ **do**
  Pose hypothesis generation
  Draw uniformly an edge and a third correspondence adjacent to both endpoints ▷ congruent triple
  $\mathbf{T}_i \leftarrow \mathrm{Procrustes}$ on the triple ▷ object frame → query camera
  Distance-color field-based cost evaluation
  $\mathbf{p}_{ij} \leftarrow \mathbf{T}_i^{-1}\mathbf{p}_j^{\mathrm{q}}$ for $j = 1, \ldots, n$ ▷ one lookup of Ω_d and Ω_c per point
  $J(\mathbf{T}_i) \leftarrow J_d(\mathbf{T}_i) + \alpha\, J_c(\mathbf{T}_i)$ ▷ Eqs. (11)–(13)
  if $J(\mathbf{T}_i) > (1+\tau)\, J^*_{\mathrm{raw}}$ **then**
    continue ▷ early stop — no refinement for this hypothesis
  end if
  $J^*_{\mathrm{raw}} \leftarrow \min\big(J^*_{\mathrm{raw}}, J(\mathbf{T}_i)\big), \quad J^* \leftarrow \min\big(J^*, J(\mathbf{T}_i)\big)$
  for L iterations **do** ▷ DCF-based ICP with per-step verification
    Recompute $\mathbf{p}_{ij}$ at $\mathbf{T}_i$ and read Ω_d, $\nabla\Omega_d$, Ω_c, $\nabla\Omega_c$
    $\boldsymbol{\delta} \leftarrow$ damped Gauss–Newton step on J linearized at $\mathbf{T}_i$ ▷ App. A.3
    $\mathbf{T}_i' \leftarrow \mathbf{T}_i\, \mathrm{Exp}(\boldsymbol{\delta})^{-1}, \quad J^* \leftarrow \min\big(J^*, J(\mathbf{T}_i')\big)$ ▷ re-evaluate the cost after every iteration
    if $J(\mathbf{T}_i') > (1+\tau)\, J^*$ **then**
      break
    end if
    if $J(\mathbf{T}_i') > J(\mathbf{T}_i)$ **then**
      break
    end if
    $\mathbf{T}_i \leftarrow \mathbf{T}_i'$
  end for
end for
Final selection and rebasing
return $\hat{\mathbf{T}}^{\mathrm{q}\leftarrow\mathrm{r}} \leftarrow \hat{\mathbf{T}}^{\mathrm{q}}(\mathbf{T}^{\mathrm{r}})^{-1}$ ▷ $\hat{\mathbf{T}}^{\mathrm{q}}$: object frame → query camera

A.3 DCF-BASED ICP OPTIMIZATION

This section details the refinement step of Sec. 3.3.3. A twist $\boldsymbol{\delta} \in \mathbb{R}^6$, which the exponential map $\mathrm{Exp} : \mathbb{R}^6 \to \mathrm{SE}(3)$ turns into a rigid motion, moves an object-frame point $\mathbf{p}_{ij}$ by $\mathbf{G}_{ij}\boldsymbol{\delta}$ to first order, with $\mathbf{G}_{ij} = [\,-[\mathbf{p}_{ij}]_\times \mid \mathbf{I}\,] \in \mathbb{R}^{3\times 6}$ and $[\cdot]_\times$ the cross-product matrix. The distance and the color read at the moved point then change by $\mathbf{a}_{ij}^\top\boldsymbol{\delta}$ and $\mathbf{J}_{ij}\boldsymbol{\delta}$, with

$$\mathbf{a}_{ij} = \mathbf{G}_{ij}^\top \frac{\nabla\Omega_d(\mathbf{p}_{ij})}{\|\nabla\Omega_d(\mathbf{p}_{ij})\|}, \qquad \mathbf{J}_{ij} = \nabla\Omega_c(\mathbf{p}_{ij})\, \mathbf{G}_{ij}, \tag{15}$$

where $\nabla\Omega_c(\mathbf{p}_{ij}) \in \mathbb{R}^{3\times 3}$ is the Jacobian of the color field. A damped Gauss–Newton step on the cost J of Eq. (13) then solves for the twist,

$$\begin{aligned}\boldsymbol{\delta} = \arg\min_{\boldsymbol{\delta}} \; & \sum_{j=1}^{n} \frac{w_{ij}^d}{\theta_d^2}\big(\Omega_d(\mathbf{p}_{ij}) + \mathbf{a}_{ij}^\top\boldsymbol{\delta}\big)^2 \\ & + \sum_{j=1}^{n} \frac{\alpha\, w_{ij}^c}{\theta_c^2}\big\|\Omega_c(\mathbf{p}_{ij}) - \mathbf{c}_j^{\mathrm{q}} + \mathbf{J}_{ij}\boldsymbol{\delta}\big\|^2 + \mu\|\boldsymbol{\delta}\|^2,\end{aligned} \tag{16}$$

where the weights $w_{ij}^d = \mathbb{1}\big[\Omega_d(\mathbf{p}_{ij}) < \theta_d\big]$ and $w_{ij}^c = \mathbb{1}\big[\|\Omega_c(\mathbf{p}_{ij}) - \mathbf{c}_j^{\mathrm{q}}\| < \theta_c\big]$ follow from the truncation in Eqs. (11) and (12), and μ damps the step relative to the mean diagonal of the normal equations. Points beyond a truncation radius therefore drop out of the normal equations, and the step

Table 13: Per-point cost for evaluation and ICP. Wall-clock is per query frame on LINEMOD with one job alone on one GPU; n is the number of frames timed.

Variant	$r(\mathbf{p})$	Correspondence	Complexity	Time (ms)
Point-to-point	$\min_{\mathbf{v}\in\mathbf{V}}\|\mathbf{p}-\mathbf{v}\|$	yes	$\mathcal{O}(NMF)$	6 037
Point-to-plane	$\|\langle\mathbf{p}-\Pi_{\mathbf{V}}(\mathbf{p}),\,\mathbf{n}_{\Pi}\rangle\|$	yes	$\mathcal{O}(NMF)$	6 057
Point-to-face	$\min_{\triangle\in\mathbf{F}} d(\mathbf{p},\triangle)$	yes	$\mathcal{O}(NMF)$	100 379
Point-to-field (ours)	$\Omega_s(\mathbf{p})$	no	$\mathcal{O}(NM)$	**324**

Table 14: Query outlier removal.

Outlier removal	n	AR
None	11,549	74.80
Statistical (Open3D)	11,393	74.69
Sphere rejection	11,333	74.85
Centroid (ours)	11,100	74.92

Table 15: Surface distance field Ω_d representation.

Ω_d	AR
Adaptive distance field	74.89
Tricubic interpolation	74.84
Gradient-augmented grid	74.76
Trilinear grid (ours)	74.92

Table 16: Descriptor network Φ.

Descriptor	AR
MiniSpinNet (Ao et al., 2023)	71.46
GeoTransformer (Qin et al., 2022)	73.67
dGeDi (Hamza et al., 2025) (ours)	74.92

Table 17: Color read-out off the surface.

Expansion	Color at	AR
Projected	$\mathbf{p}-s\,\nabla\Omega_s(\mathbf{p})$	74.73
Normal-fade	$\mathbf{p}$, weight $e^{-s^2/2\sigma^2}$	74.92
Plain (ours)	$\mathbf{p}$	74.92

Table 18: Color field construction and color space.

Constructed color	Color space	AR	Time(ms)
Per-vertex (barycentric)	RGB	74.93	290
Face color + gradient	RGB	74.83	330
Face color	Chroma	75.21	301
Face color	Log-chroma	75.58	303
Face color (ours)	RGB	74.92	267

reduces to least squares over exactly the points that agree under the cost. The pose is then updated as $\mathbf{T}_i \leftarrow \mathbf{T}_i\,\mathrm{Exp}(\boldsymbol{\delta})^{-1}$.

A.4 Per-point Cost of the Evaluation and ICP

Table 7 reports accuracy and wall-clock for each per-point cost; this section accounts for the wall-clock column. Verification evaluates a residual at every query point under every hypothesis: with a query cloud of M points and a pool of N hypotheses, each hypothesis maps the query into the object frame, $\mathbf{p} = T^{-1}\mathbf{q}$, so the residual is evaluated at NM positions per frame. Table 13 gives the resulting cost, where $\mathbf{V}$ and $\mathbf{F}$ are the vertices and triangles of $\mathcal{O}$ and $\mathbf{n}_{\Pi}$ is the normal at the nearest vertex. The three classical residuals resolve a correspondence at query time and therefore visit every element of the model, carrying the factor F into the inner loop; the field is read by trilinear interpolation after a single $\mathcal{O}(R^3)$ bake that the whole pool shares, which removes that factor. Point-to-point and point-to-plane agree to within 0.3% because both reduce to one squared distance per pair, the latter adding a single inner product once the nearest vertex is known, while point-to-face must locate the closest point on the triangle itself. The three point-to-field rows of Table 7 share the $\mathcal{O}(NM)$ cost and differ only in which terms of the field are read, which is why they lie within 25 ms of each other. The classical entries are lower bounds: the model is subsampled to $F = 2048$ faces, and the substitution is applied to verification only, refinement retaining the field lookup.

A.5 Additional Ablations

We further ablate the components not covered in Sec. 4.4.

Table 19: Points m and n.

	512	1024	2048	4096
m	70.23	74.04	74.92	74.81
n	72.26	74.69	74.92	74.86

Table 20: Color weight α of the cost.

α	0	0.1	0.25	0.5	1.0
AR	73.94	74.04	74.92	73.20	70.46

Table 21: Resolution R.

R	AR	Memory(MB)
32	74.17	0.5
64	74.92	4.0
128	74.97	32.0

Table 22: Hypotheses N.

N	AR	Time(ms)
1k	73.02	277
5k	74.54	319
10k	74.69	343
20k	74.92	384
40k	74.86	480

Table 23: ICP iterations L.

L	AR	Time(ms)
2	74.81	268
4	74.87	285
8	74.92	310
12	74.96	341
16	74.98	386

Query outlier removal. Stray points from depth noise and mask errors enter the cost of every hypothesis through Eqs. (11) and (12), and we therefore compare ways of removing them from the query point cloud $\mathbf{P}^{\mathrm{q}}$ (Tab. 14): no removal, the statistical outlier removal of Open3D, a sphere rejection, and our removal of points beyond the mean plus 1.5 standard deviations of their distance to the centroid. The centroid-based removal discards the most points and yields the best accuracy, whereas statistical outlier removal falls even below no removal.

Surface distance field. As Ω_d of Eq. (8) is read at every query point of every hypothesis in Eq. (11) and differentiated during refinement in Eq. (15), we compare its representation (Tab. 15): an adaptive distance field, tricubic instead of the trilinear interpolation of Eq. (7), a grid augmented with stored gradients, and our trilinear grid. The trilinear grid yields the best accuracy, and neither adaptive refinement nor higher-order or gradient-augmented interpolation improves on it, presumably because the truncation in Eq. (11) makes the cost insensitive to small interpolation errors.

Descriptor network. The descriptors determine the similarity matrix $\mathbf{S}$ of Eq. (5) and thus the correspondence set $\mathcal{C}$ of Eq. (6), and we compare registration descriptors, MiniSpinNet (Ao et al., 2023), GeoTransformer (Qin et al., 2022), and dGeDi (Hamza et al., 2025) as Φ (Tab. 16). dGeDi, distilled for 6D pose estimation, yields the best accuracy among the evaluated descriptors.

Color field construction. We compare how ν_c of Eq. (9) assigns a color to each grid point and in which color space the color cost of Eq. (12) is computed (Tab. 18): per-vertex colors interpolated barycentrically at $\Pi(\mathbf{x})$, the face color with a color gradient, and our constant face color, the last in RGB, chroma, and log-chroma. Per-vertex interpolation and color gradients bring no gain over the constant face color while increasing the runtime, whereas the chroma and log-chroma spaces, which discount illumination, improve the accuracy further.

Color read-out. As a query point rarely lies exactly on the surface, we compare where the color in Eq. (12) is read (Tab. 17): at the point projected onto the surface along the distance gradient, $\mathbf{p} - \Omega_d(\mathbf{p})\nabla\Omega_d(\mathbf{p})$, at the point with a weight $e^{-\Omega_d(\mathbf{p})^2/2\sigma^2}$ that fades the color term with the distance, and directly at the point, $\Omega_c(\mathbf{p})$. Reading the color directly at the point is as accurate as fading it with the distance and more accurate than the projection, and we therefore use this simplest read-out.

A.6 Hyper-parameter Studies

We vary one hyper-parameter at a time around the values of Sec. 4.2.

Number of points. The numbers of reference and query points m and n used for the descriptors set how many correct matches $\mathcal{C}$ of Eq. (6) can contain, while the size of $\mathbf{S}$ in Eq. (5) and the runtime grow with their product (Tab. 19). Fewer points miss matches on small or thin parts, and more points add ambiguous matches at a higher cost, and $m = n = 2048$ gives the best accuracy.

Color weight. The weight α balances the color cost of Eq. (12) against the distance cost of Eq. (11) in J of Eq. (13) (Tab. 20). Without color, the cost cannot separate poses that fit the geometry equally well, whereas a large weight lets illumination changes and errors in the reconstructed texture dominate, and $\alpha = 0.25$ gives the best accuracy.

Resolution. The lattice resolution R of Eq. (7) sets how finely Ω_d and Ω_c of Eqs. (8) and (9) resolve the surface, and the memory of the fields grows cubically with it (Tab. 21). A coarse lattice blurs thin parts and color boundaries, whereas a finer lattice improves the accuracy only marginally at eight times the memory, and $R = 64$ offers the best trade-off.

Pool size. The number of hypotheses N bounds how many congruent triples under Eq. (10) are drawn and thus the chance of drawing one close to the true pose (Tab. 22). Too few hypotheses miss the correct pose among many outlier correspondences, whereas more hypotheses mainly add runtime, and $N = 20{,}000$ gives the best accuracy.

Iterations. The maximum number of ICP iterations L, each a step of Eq. (16), limits how far a hypothesis can move during refinement (Tab. 23). Too few iterations leave coarse hypotheses unconverged, whereas more iterations add runtime once the cost has converged, and $L = 8$ gives the best accuracy.

A.7 Additional Qualitative Results

Fig. 4 and Fig. 5 presents additional qualitative results across various objects and viewpoints.

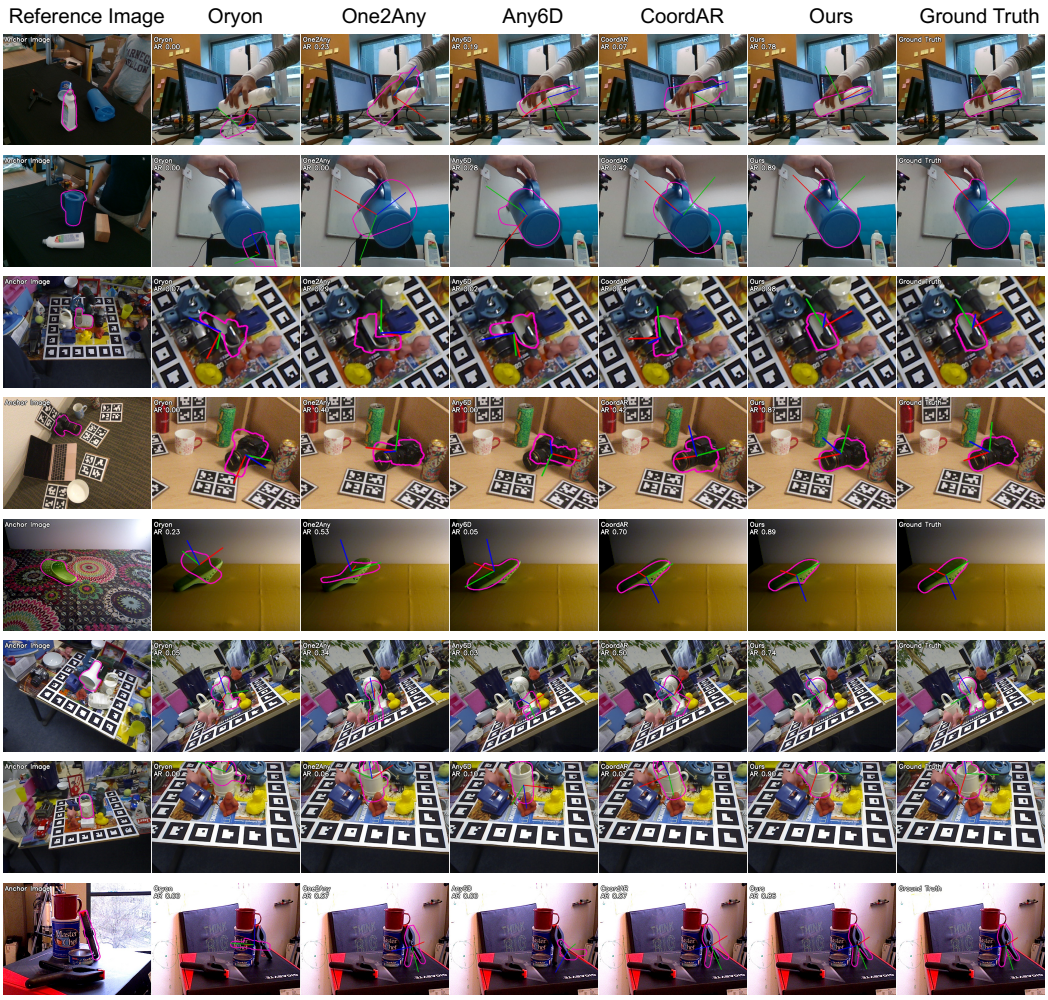

Figure 4: **Additional qualitative results.**

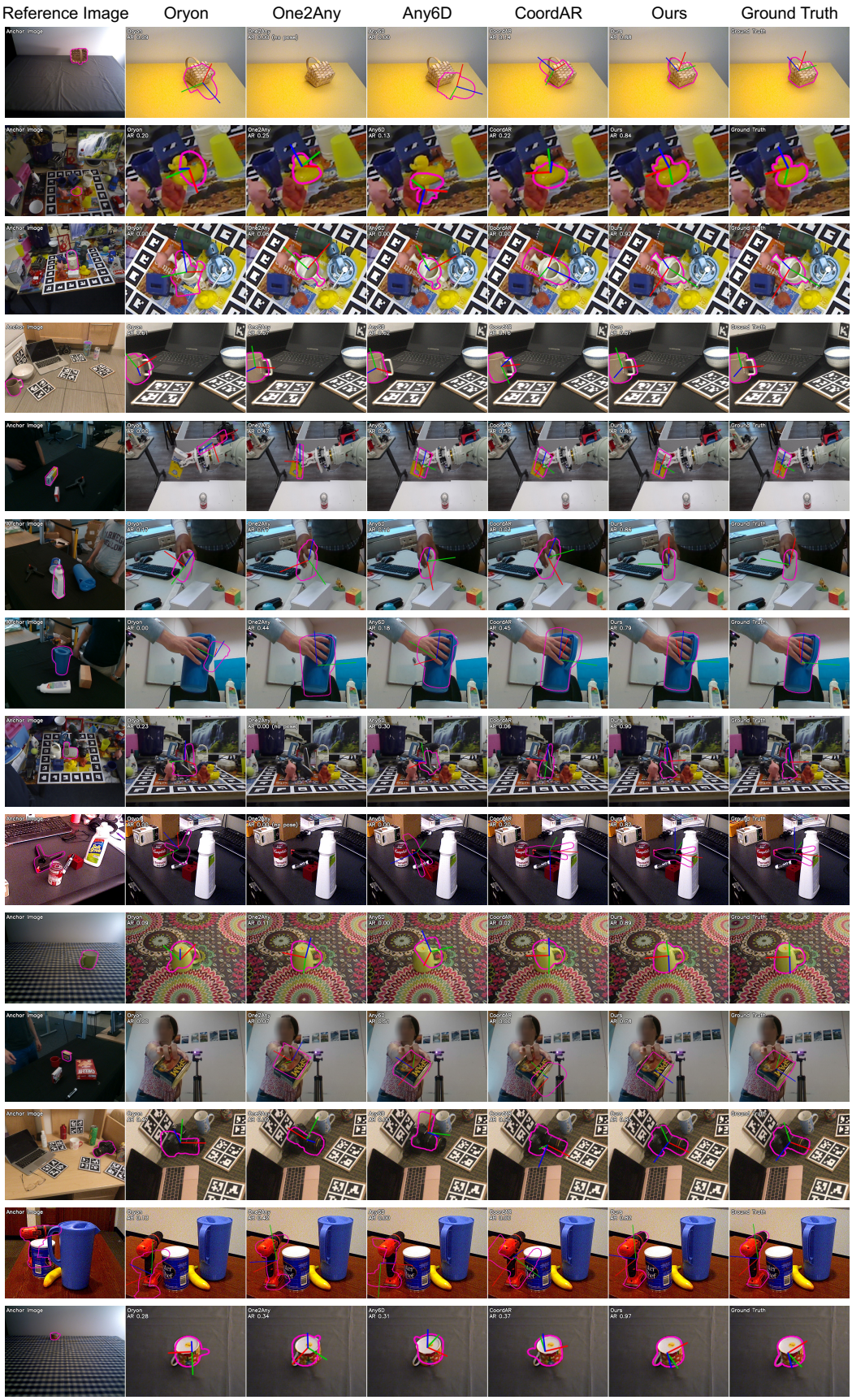

Figure 5: **Additional qualitative results.**