HaNDF: Object-Conditioned Geometric Neural Hand Distance Fields

Zuhao Liu **Melih Darcan** **Zhengdi Yu** **Riza Alp Guler** **Tolga Birdal**
Imperial College London

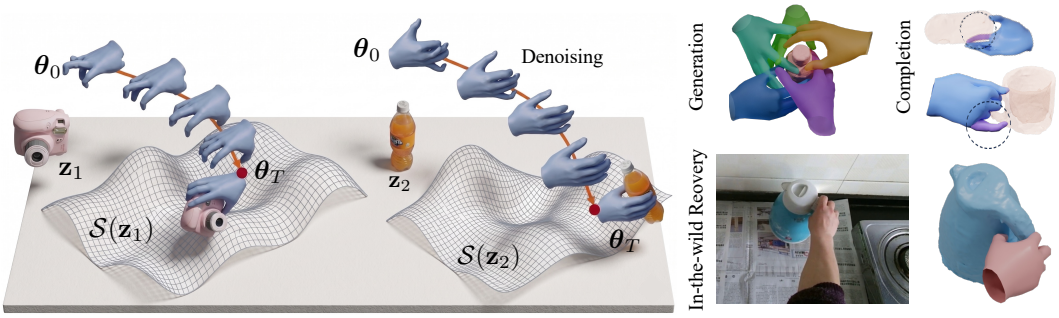

Figure 1: We introduce **HaNDF**, an object-conditioned generative hand prior. HaNDF is a geometric neural distance field representing the plausible hand configurations under interaction, and enables numerous tasks from denoising and completion to recovering hands or even objects from image data.

Abstract

We propose, **HaNDF**, object-conditional neural distance fields as data-driven priors for modeling the plausible hand–object configurations, in the product manifold of hand articulations. While recent unconditional generative priors showed success in modeling human pose, hands are versatile articulated bodies often found in interaction with other objects. This makes it crucial to develop a generative prior which can model the plausible hand poses conditioned on the object under interaction. To this end, our HaNDF proposes a geometry-aware, object-conditioned neural distance field, representing the plausible articulations in the zero-level set of a conditional field. We leverage tools from Riemannian geometry to (i) *represent* plausible hands in the zero level-set of an object-induced neural field, and (ii) *project* a given pose onto this field. Using HaNDF, we can also optimize for a plausible *condition*, determining the object under interaction. Our extensive evaluations demonstrate that our HaNDF provides a flexible and powerful prior for hand–object interaction, suitable for applications such as pose refinement, reconstruction under occlusion, and physically plausible manipulation synthesis.

1 Introduction

Human hands are among the most versatile articulated systems, practicing rich and diverse interactions with objects. Modeling plausible hand–object configurations is therefore fundamental to applications in robotics, AR/VR, and 3D perception. Yet, recovering or generating such interactions from often partial observations remains challenging: the space of valid configurations is highly constrained, object-dependent, and non-Euclidean, where even small pose perturbations can lead to physically implausible grasps with interpenetration or loss of contact. A central requirement is thus a strong, queryable prior that encodes which hand poses are plausible given an object.

Existing approaches fall short of this goal. Generative models based on VAEs [40] or diffusion [42, 16] learn distributions over poses or interactions, but operate in Euclidean latent spaces where distances lack geometric meaning, limiting their expressivity. Neural distance field [14] (NDF) variants such as

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

PoseNDF [45] or NRDF [22] offer an alternative by modeling the manifold of plausible articulations directly and enabling projection via gradient descent. However, current NDF-based priors are human-body specific, unconditional, capture hand articulation independent of the object, and thus fail to encode interaction-specific constraints. Conversely, object-conditioned methods [28, 48] typically treat articulation in Euclidean space or rely on task-specific pipelines, sacrificing geometric consistency. As a result, no existing approach provides a unified prior that is (i) geometry-aware, (ii) conditioned on object shape, and (iii) equipped with a meaningful distance for optimization.

HaNDF addresses this gap proposing a generative, object conditioned prior of hand poses. Our key insight is to model hand articulation on its intrinsic Riemannian manifold while treating the object as a conditioning signal that modulates the manifold of plausible configurations. Concretely, we learn a conditional distance field that predicts the geodesic distance to the nearest valid hand pose given object geometry, whose zero level set defines the object-specific interaction manifold. This formulation yields a continuous, geometry-aware energy landscape where Riemannian gradients provide principled projection, refinement, and sampling directly on the articulation manifold. This perspective unifies multiple applications under a single framework without retraining: denoising, recovering physically consistent grasps from data, and generation of novel grasps. When integrated into reconstruction pipelines, HaNDF acts as a plug-and-play prior, improving robustness under occlusion without retraining task-specific models, as validated across three challenging benchmarks: HOI4D [32], OakInk [50], and DexYCB [8]. In summary, **our contributions are**:

- We introduce HaNDF, an object-conditioned Riemannian neural distance field for hand articulation, whose zero-level set defines a family of object-dependent plausibility manifolds.
- We develop an object-specific distance-field training procedure with per-object latent fitting and nearest-pose supervision, together with a transformer architecture that jointly processes hand-joint and object tokens.
- Based upon NRDF's RDF-Grad, we enable object-conditioned projection, denoising, completion, and generation of MANO hand poses, and introduce an inversion procedure to retrieve and refine compatible object conditions.

HaNDF consistently outperforms state-of-the-art baselines, particularly in challenging settings with occlusion and ambiguity. Our code is publicly available at circle-group.github.io/research/HaNDF.

2 Related Work

Generative priors of humans and hands. Early efforts learned plausible-pose priors from 2D data with weak 3D supervision [36], used GANs [25], or extended adversarial supervision to motion sequences [27]. Generative priors have been built on a range of architectures. PoseD-Flow [35] trains a flow matching model of human pose. MotionVAE [30], HuMoR [40], and PhaseMP [43] use VAEs autoregressively for motion priors. Diffusion models [44, 42, 17] have since become standard for pose and motion generation, with recent extensions to whole-body poses including hands [33]. DynHaMR [55] trains an HMP prior for hand-motion infilling. For hand-object interaction, TOCH [59] learns an object-centric correspondence-field prior for refining tracked sequences. Object-conditional grasp generators form a parallel literature [24, 51, 7, 60, 29]. These methods generate diverse poses but typically lack a differentiable distance to the pose manifold. We instead learn a geodesic distance to the conditioned plausible-grasp set, whose Riemannian gradient yields an projection direction.

Neural distance fields (NDFs). NDFs were introduced for implicit representation of 3D objects and scenes [14] and have since become a tool for learning priors over articulated forms. PoseNDF [45] models body articulation as the zero-level set of an NDF. NRDF [22] introduces a geometrically principled variant on the product of manifolds with better accuracy and diversity. NRMF [53] extends NRDF to temporal motion; Gu *et al.* [20] train NRDF on AMASS Retargeted-G1 [34] for humanoid retargeting. We extend the conditional setting to human hands–objects, with a Riemannian distance field on the product manifold of joint quaternions and a learned object shape latent as condition.

Hand-object interaction reconstruction. Hand-object reconstruction systems jointly recover hand and object geometry from monocular images. Early parametric pipelines couple MANO with object pose regression [21]. SDF-based methods replace meshes with implicit fields: Grasping Field [26], AlignSDF [12], gSDF [11], and HOISDF [39]. Recent approaches leverage large pretrained models: diffusion priors for object completion under occlusion [31, 19, 46, 1, 51] and feed-forward transformers for direct point-cloud prediction [13]. Our contribution is a hand prior, not a reconstruction

pipeline: a differentiable distance to the conditioned plausible manifold whose gradient supports optimisation, denoising, completion, and refinement on top of any of the systems above.

3 Background

Since hand joints are rotations, their natural configuration space is not Euclidean but a product of rotations. Here we focus on unit-quaternion representation of rotations and review the differential / Riemannian geometry of rotations and the product-manifold structure in Sec. A. Our exposition is based on NRDF [22] and NRMF [53], akin to [5, 4, 9, 6]. In what follows, let $\mathcal{M}$ be a smooth m-dimensional manifold embedded in an ambient Euclidean space $\mathcal{X} = \mathbb{R}^d$. For $\mathbf{x} \in \mathcal{M}$, the *tangent space* $T_{\mathbf{x}}\mathcal{M}$ is the set of all velocity vectors of smooth curves passing through $\mathbf{x}$.

The rotation group as a Riemannian manifold. The space of 3D rotations is a particular Riemannian manifold, known as the *special orthogonal group*:

$$\mathrm{SO}(3) = \left\{ \mathbf{R} \in \mathbb{R}^{3\times 3} \;\middle|\; \mathbf{R}^\top \mathbf{R} = \mathbf{I},\ \det(\mathbf{R}) = 1 \right\}. \tag{1}$$

Inspired by [22], in this work, we parameterize rotations by *unit quaternions* (UQ) and provide a detailed review of the group $\mathrm{SO}(3)$ in Sec. B.

A quaternion is an element $\mathbf{q} = (q_w, q_x, q_y, q_z) = (w, \mathbf{v}) \in \mathbb{R}^4$, with scalar part $w \in \mathbb{R}$ and vector part $\mathbf{v} \in \mathbb{R}^3$. The conjugate is $\bar{\mathbf{q}} = (w, -\mathbf{v})$, with the norm $\|\mathbf{q}\|_2 = \sqrt{w^2 + \|\mathbf{v}\|_2^2}$. The *Hamilton product* of $\mathbf{q} = (w_\mathbf{q}, \mathbf{v}_\mathbf{q})$ and $\mathbf{r} = (w_\mathbf{r}, \mathbf{v}_\mathbf{r})$ is

$$\mathbf{q} \otimes \mathbf{r} = \left(w_\mathbf{q} w_\mathbf{r} - \mathbf{v}_\mathbf{q}^\top \mathbf{v}_\mathbf{r},\ w_\mathbf{q}\mathbf{v}_\mathbf{r} + w_\mathbf{r}\mathbf{v}_\mathbf{q} + \mathbf{v}_\mathbf{q} \times \mathbf{v}_\mathbf{r} \right). \tag{2}$$

Shown in Fig. 2, the unit quaternions form the three-sphere:

$$\mathcal{M}_\mathbb{H} = \mathbb{H}_1 = \left\{ \mathbf{q} \in \mathbb{R}^4 \;\middle|\; \|\mathbf{q}\|_2 = 1 \right\} \simeq S^3, \tag{3}$$

and form a Lie group under Hamilton product. The inverse of a UQ is its conjugate: $\mathbf{q}^{-1} = \bar{\mathbf{q}}$. For a rotation by angle α around unit axis $\mathbf{u} \in S^2$, the corresponding UQ is

$$\mathbf{q} = (\cos(\alpha/2),\ \mathbf{u}\sin(\alpha/2)). \tag{4}$$

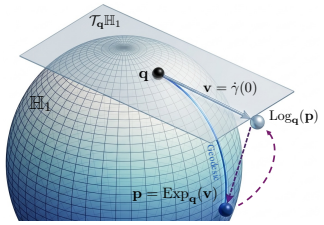

Figure 2: Manifold of quaternions.

The map from $\mathbb{H}_1$ to $\mathrm{SO}(3)$ is a two-to-one (double) covering map: $\mathbf{q} \sim -\mathbf{q}$. That is, $\mathbf{q}$ and $-\mathbf{q}$ represent the same physical rotation. The tangent space at $\mathbf{q}$ is

$$T_\mathbf{q}\mathbb{H}_1 = \left\{ \boldsymbol{\eta} \in \mathbb{R}^4 \;\middle|\; \mathbf{q}^\top \boldsymbol{\eta} = 0 \right\}. \tag{5}$$

The orthogonal projection from ambient space $\mathbb{R}^4$ onto $T_\mathbf{q}\mathbb{H}_1$ is

$$\Pi_\mathbf{q}^{S^3}(\mathbf{v}) = \mathbf{v} - (\mathbf{q}^\top \mathbf{v})\mathbf{q} = (\mathbf{I} - \mathbf{q}\mathbf{q}^\top)\mathbf{v}. \tag{6}$$

For a smooth $f : \mathbb{H}_1 \to \mathbb{R}$, if $\nabla_\mathbf{q} f$ is the Euclidean gradient, the *spherical Riemannian gradient* is

$$\mathrm{grad}_\mathbf{q} f = (\mathbf{I} - \mathbf{q}\mathbf{q}^\top)\nabla_\mathbf{q} f. \tag{7}$$

Let $\boldsymbol{\eta} \in T_\mathbf{q}\mathbb{H}_1$. In the quaternion Lie-group notation, tangent vectors are represented as pure imaginary quaternions $\boldsymbol{\eta} = (0, \mathbf{u})$ with $\mathbf{u} \in \mathbb{R}^3$. The **exponential** and **logarithmic maps** at $\mathbf{q}$ are:

$$\mathrm{Exp}_\mathbf{q}(\boldsymbol{\eta}) = \mathbf{q} \otimes \exp(\boldsymbol{\eta}) = \mathbf{q} \otimes \left(\cos\vartheta,\ \frac{\sin\vartheta}{\vartheta}\mathbf{u} \right), \quad \mathrm{Log}_\mathbf{q}(\mathbf{p}) = \left(0,\ \frac{\mathbf{v}}{\|\mathbf{v}\|_2} \cos^{-1}(w) \right), \tag{8}$$

where $\boldsymbol{\eta} = (0, \mathbf{u})$ and $\vartheta = \|\mathbf{u}\|_2$, and the *group exponential at the identity* is

$$\exp(0, \mathbf{u}) = \left(\cos\|\mathbf{u}\|_2,\ \frac{\sin\|\mathbf{u}\|_2}{\|\mathbf{u}\|_2}\mathbf{u} \right), \tag{9}$$

with the convention $\sin\|\mathbf{u}\|_2/\|\mathbf{u}\|_2 = 1$ when $\|\mathbf{u}\|_2 = 0$ and $\mathrm{Log}_\mathbf{q}$ is zero when $\|\mathbf{v}\|_2 = 0$, as used in NRDF [22]. The **Riemannian** (**geodesic**) distance between two unit quaternions $\mathbf{q}_1, \mathbf{q}_2 \in \mathbb{H}_1$ is the length of the shortest geodesic on S^3, accounting for the antipodal identification $\mathbf{q} \sim -\mathbf{q}$:

$$d_{\mathbb{H}_1}(\mathbf{q}_1, \mathbf{q}_2) = 2\cos^{-1}\left(|\mathbf{q}_1^\top \mathbf{q}_2| \right) = 2\min\left\{ \|\mathrm{Log}_{\mathbf{q}_1}(\mathbf{q}_2)\|_2,\ \|\mathrm{Log}_{\mathbf{q}_1}(-\mathbf{q}_2)\|_2 \right\}. \tag{10}$$

4 HaNDF: Geometric Object Conditioned Neural Hand Distance Fields

Hand representation. We instantiate the articulated hand with the MANO model [41], which represents a hand mesh as a differentiable function of shape, pose, and global rigid motion:

$$\mathbf{V} = \mathcal{H}(\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{t}) \in \mathbb{R}^{N_v \times 3}, \tag{11}$$

where $\boldsymbol{\beta} \in \mathbb{R}^{N_\beta}$ denotes low-dimensional hand shape, $\mathbf{t} \in \mathbb{R}^3$ is global translation, and $\boldsymbol{\theta}$ denotes the 3D articulated hand pose. In MANO, the pose consists of local rotations along a kinematic tree of hand joints. We exclude the global wrist rotation from the intrinsic articulation prior and parameterize the remaining $K = 15$ local joint rotations on the power manifold of $\boldsymbol{\theta} := \{\mathbf{q}_i \in \mathbb{H}_1\}_{i=1}^K \in \mathcal{M}_H = \mathbb{H}_1^K$. Each $\mathbf{q}_i$ parameterizes the rotation of joint i with respect to its parent in the MANO kinematic tree. This *relative*, *local* parameterization is important in coupling the configuration of all local rotations.

Object representation. Let $\mathcal{O}$ denote the space of 3D objects, represented as meshes, point clouds, or implicit fields. We assume access to an encoder $E : \mathcal{O} \to \mathcal{Z}$, which maps an object $O \in \mathcal{O}$ to a latent code: $\mathbf{z} = E(O) \in \mathcal{Z}$. Unlike the hand pose, $\mathcal{Z} \equiv \mathbb{R}^{d_z}$ is not endowed with a special Riemannian structure. The encoder is trained jointly with a decoder $D : \mathcal{Z} \to \mathcal{O}$ in an autoencoding framework. As we will detail in Sec. 4.1, while any embedding network is possible, we adopt a DeepSDF-style [37] architecture that produces a compact latent code capturing global object geometry.

We deliberately represent the object via a latent code rather than explicit geometry (*e.g.*, point clouds or meshes) for reasons of: (i) *compactness*, enabling efficient conditioning in neural fields; (ii) *invariance* to input resolution and representation; (iii) *compatibility* with generative and reconstruction pipelines, where object shape may be partially observed.

4.1 HaNDF

Given a hand pose $\boldsymbol{\theta} \in \mathcal{M}_H$ and an object latent $\mathbf{z} \in \mathcal{Z}$, our model learns a scalar distance field

$$f_\Phi : \mathcal{M}_H \times \mathcal{Z} \to \mathbb{R}_{\geq 0}, \tag{12}$$

that explicitly approximates the geodesic distance to the nearest plausible hand pose conditioned on the object. The *conditioned manifold* of plausible hands is thus given by the zero level-set:

$$\mathcal{S}(\mathbf{z}) = \{\boldsymbol{\theta} \in \mathcal{M}_H \mid f_\Phi(\boldsymbol{\theta} \mid \mathbf{z}) = 0\}, \tag{13}$$

such that the value of f represents the unsigned geodesic distance to the closest plausible pose on the manifold. The latent $\mathbf{z}$ influences the field globally, adapting the location and shape of the zero level-set $S(z)$ according to object geometry. Intuitively, $\mathbf{z}$ encodes affordances such as size, curvature, and topology, which determine feasible grasp configurations.

Projection onto HaNDF. Given an object $\mathbf{z}$, an arbitrary hand pose $\boldsymbol{\theta}^{(0)} \in \mathcal{M}_H$ can be projected onto the closest plausible hand pose on our object-conditioned manifold:

$$\Pi_{\mathcal{S}(\mathbf{z})}(\boldsymbol{\theta}^{(0)}) = \operatorname*{arg\,min}_{\boldsymbol{\theta} \in \mathcal{S}(\mathbf{z})} d_{\mathcal{M}_H}(\boldsymbol{\theta}, \boldsymbol{\theta}^{(0)}). \tag{14}$$

Since $\mathcal{S}(\mathbf{z})$ is represented implicitly by the zero level-set of f_Φ, we solve this optimization by minimizing the predicted distance field over the hand manifold, similar to [45, 22]:

$$\boldsymbol{\theta}^\star \approx \operatorname*{arg\,min}_{\boldsymbol{\theta} \in \mathcal{M}_H} f_\Phi(\boldsymbol{\theta} \mid \mathbf{z}). \tag{15}$$

Since $\mathbf{z}$ is a condition, the optimization is performed only over $\boldsymbol{\theta}$. We then update the hand pose using an *adaptive conditioned Riemannian descent step* (see Fig. 1 for iterations):

$$\boldsymbol{\theta}^{(t+1)} = \operatorname{Exp}_{\boldsymbol{\theta}^{(t)}} \left(-\alpha f_\Phi(\boldsymbol{\theta}^{(t)} \mid \mathbf{z}) \frac{\operatorname{grad}_{\boldsymbol{\theta}} f_\Phi(\boldsymbol{\theta}^{(t)} \mid \mathbf{z})}{\left\| \operatorname{grad}_{\boldsymbol{\theta}} f_\Phi(\boldsymbol{\theta}^{(t)} \mid \mathbf{z}) \right\|_{\boldsymbol{\theta}^{(t)}} + \epsilon} \right), \tag{HaNDFGrad}$$

where $\alpha > 0$ is a step-size parameter and $\epsilon > 0$ prevents numerical instability. By construction, Eq. (HaNDFGrad) respects the non-Euclidean manifold of MANO hand articulations as every update remains on $\mathcal{M}_H = \mathbb{H}_1^K$, and the multiplication by $f_\Phi(\boldsymbol{\theta}^{(t)} \mid \mathbf{z})$ makes the step length proportional to the predicted distance, akin to the RDFGrad of NRDF [22]. The projection is terminated when either the predicted distance falls below a threshold $f_\Phi(\boldsymbol{\theta}^{(t)} \mid \mathbf{z}) < \delta$ or maximum iterations is reached.

HaNDF as a conditional generative model. Beyond projection and refinement, HaNDF can be used as a conditioned generative model of plausible hand poses. For a fixed latent $\mathbf{z}$, the f-induced Bolzmann density $p(\boldsymbol{\theta} \mid \mathbf{z}) \propto \exp(-\beta f_\phi(\boldsymbol{\theta} \mid \mathbf{z})^2)$, approximates data distribution, understood as a product-measure over spheres, illustrated in Fig. 3, analogous to [53]. Eq. (HaNDFGrad) is then a maximum-aposteriori estimation.



Figure 3: HaNDF jointly models the plausible configurations of joints given the object code.

Our simple strategy to sample from $p(\cdot \mid \mathbf{z})$ is as follows: we (i) initialize a random hand pose $\boldsymbol{\theta}^{(0)}$ by sampling each quaternion independently, $\mathbf{q}_k^{(0)} \sim \mathrm{Unif}(\mathbb{H}_1)\,\forall k$ or sampling $\tilde{\mathbf{q}}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_4)$ and normalizing $\mathbf{q}_k^{(0)} = \tilde{\mathbf{q}}_k / \|\tilde{\mathbf{q}}_k\|_2$, (ii) project it onto HaNDF, and (iii) pass it from MANO model in Eq. (11). Different initial samples can project to different connected components or basins of attraction of $\mathcal{S}(\mathbf{z})$. Hence, generation diversity is controlled by the initial distribution and by the geometry of the learned field.

A connection to Riemannian Flow Matching [10] has been made in [22] for the unconditional NRDF. Our distance field formulation has some advantages. First, it avoids imposing a Gaussian structure on the hand-poses. Instead of predicting the step towards the manifold via a network, we obtain it via backpropagation. This allows optimization in domains where designing decoders is challenging. Also, we can utilize existing optimizers of deep learning frameworks for Lagrangian iterations by simply minimizing distance.

HaNDF as a generative prior. The same learned field can be used for generation, denoising, completion, and reconstruction, since all tasks reduce to minimizing or projecting through the same object-conditioned distance field. Typically, in downstream tasks, our prior is combined with task-specific data terms involving observations $\mathcal{Y}$, leading to the inference problem[1]:

$$\min_{\boldsymbol{\theta} \in \mathcal{M}_H, \mathbf{z} \in \mathcal{Z}} \quad \mathcal{L}_{\mathrm{data}}(\boldsymbol{\theta}, \mathbf{z} \mid \mathcal{Y}) + \lambda_{\mathrm{prior}} f_\Phi(\boldsymbol{\theta} \mid \mathbf{z})^2 + \lambda_z \mathcal{R}(\mathbf{z}), \tag{16}$$

which can be minimized for $\boldsymbol{\theta} \in \mathcal{M}_H$ (Riemannian Adam [2]) to recover hands or for $\mathbf{z}$ (Euclidean) to recover objects that best fit the hand. The regularizer $\mathcal{R}(\mathbf{z})$ prevents degenerate latent drift and may be given by a reconstruction loss through the object decoder. $\mathcal{L}_{\mathrm{data}}$ is task-dependent and can ensure consistency with observed depth, silhouette, 3D measurements.

Object encoding. We represent the object as a single global shape code $\mathbf{z} \in \mathbb{R}^{c_o}$ produced by a pretrained DeepSDF auto-decoder [37]. Following DeepSDF, each shape is paired with an continuous latent code $\mathbf{z}$, and the decoder $g_\psi(\mathbf{z}, \mathbf{x})$ regresses the signed distance at any query point. We choose DeepSDF auto-decoder for two reasons: (i) its latent space is inherently aligned with our object condition formulation, and (ii) its decoder-only design supports geometry refinement via latent optimization at test time. Note that our object guidance only requires a permutation-invariant representation of the object and is therefore compatible with alternative encoders [15, 49].

To recover objects at inference, we invert (auto-decode) to obtain the DeepSDF latent code K times, starting from different initializations and retrieve the nearest training latents, and keep hand-object pair with smallest error. We use PCA to reduce the dimensionality of this search. We provide the details and comparisons to other encoders in Sec. C.3.

HaNDF architecture. We tokenize hand and object into a shared d-dimensional space and process them with a transformer encoder. The K joint quaternions are linearly projected to hand tokens $\mathbf{h}_k = \mathbf{W}_h \mathbf{q}_k + \mathbf{b}_h$. We project the DeepSDF latent code to obtain one object token $\mathbf{o} = \mathbf{W}_o \mathbf{z} + \mathbf{b}_o$. A learnable `[DIST]` token $\mathbf{d}$ is prepended as an aggregator, yielding the input sequence

$$\mathbf{X} \;=\; [\,\mathbf{d};\; \mathbf{h}_1, \ldots, \mathbf{h}_K;\; \mathbf{o}\,] \;\in\; \mathbb{R}^{(K+2)\times d}. \tag{17}$$

After learnable positional embeddings, $\mathbf{X}$ is processed by L self-attention layers; the final-layer representation at the `[DIST]` position is mapped to the scalar prediction by a linear head with softplus function,

$$f_\Phi(\boldsymbol{\theta} \mid \mathbf{z}) \;=\; \mathrm{softplus}(\mathbf{w}^\top \mathbf{d}^{(L)} + b). \tag{18}$$

Self-attention lets every hand-joint token attend to the object token and vice versa, so the prediction is jointly conditioned on hand articulation and object shape. Encoding the entire shape code as a

[1] In the language of diffusion or flow matching, such inference is understood as *inversion* or *training-free guidance* [35, 3].

single token preserves its holistic semantics and lets each joint query the full object descriptor in one attention hop; we use $L = 4$ layers and $c_o = 64$, with both choices justified in Tab. 12. MLP is also an alternative to build the architecture, as shown in Sec. C.1.

Training. We train f_Φ using the regression loss on paired hand–object data with synthetic noise applied to the hand pose. The training distribution must place sufficient mass on near-manifold (hard-negative) samples since at inference our projection spends most of its iterations around the *narrow band*. Hence, we inherit the exponential sampling schedule from [22] for hands. The exponential distribution assigns higher density to small magnitudes than to large ones, with non-zero density throughout the tail. We also sample the conditioning object over the epochs proportional to the plausible grasps per object provided in the dataset. We provide the full training algorithm in Alg. 1.

5 Experiments

To evaluate the effectiveness of the proposed HaNDF, we conduct experiments on generation, denoising, in-the-wild hand recovery, and hand–object recovery, using task-specific evaluation metrics. We evaluate our method on several datasets, including HOI4D [32], OakInk [50], and DexYCB [8]. We also provide detailed ablation studies on the network architecture, object encoding, and field geometry, along with further analyses of runtime, robustness, and generalization.

Datasets. We evaluate our method on three large-scale hand–object interaction datasets. **HOI4D** [32] provides egocentric RGB-D sequences with dense MANO annotations, object poses, and masks. We follow an object-level split across six rigid categories, using 219 objects (63K poses) to train and 30 unseen objects (7.5K poses) to test. We retain only contact frames. **DexYCB** [8] contains 1,000 third-person grasp sequences over 21 YCB objects with full MANO and object pose annotations, from which we use the standard `s0` split (582K frames). **OakInk** [50] complements these with diverse hand–object interactions in multi-view RGB sequences. We use sequences from 7 object categories to build our training and test sets, with 53 objects for training and 14 unseen objects for testing. More details in Sec. F.

Baselines. We compare against three groups of baselines. First, we adapt pose distance-field priors to the hand domain by re-implementing PoseNDF [45] and NRDF [22] on the 15-joint MANO manifold, denoted H-PoseNDF (Euclidean updates with quaternion normalization) and H-NRDF (Riemannian formulation on $(\mathbb{S}^3)^{15}$), both of which are unconditional and serve as direct ablations of our object conditioning. Second, we compare to recent hand and hand–object generative priors, including HMP [18] and object-conditioned models such as G-HOP [51], LatentHOI [29], and Text2HOI [7]. Third, for single-image hand recovery, we evaluate against EasyHOI [31], AlignSDF [12], and IHoI [52], providing ground-truth object geometry to all methods to isolate hand reconstruction, and integrating our method as an object-conditioned refinement prior within EasyHOI and AlignSDF. Sec. E provides further information.

Metrics. We adopt a unified set of evaluation metrics across all downstream tasks, covering four complementary aspects: realism, distribution fidelity, sample diversity, and pose reconstruction accuracy. All angular metrics are computed on the product manifold $(\mathbb{S}^3)^{15}$ formed by the 15 MANO finger-joint quaternions and reported in degrees; Euclidean metrics are reported in millimeters (mm) or centimeters (cm) as indicated, and F-scores are reported at distance thresholds expressed as percentages of the object's longest bounding-box half-side. A summary of our metrics are given below and Sec. G provides further details:

- Δ_q: Geodesic rotation error with GT hand across all 15 joints (degrees)
- **dNN**: Geodesic distance to nearest GT pose in train set under the same object catogory (degrees)
- **m2m**: Mean Euclidean distance between predicted and ground-truth 3D vertices (mm)
- **MPVPE / MPJPE**: Mean per-vertex / per-joint position error on MANO mesh (mm)
- **I.V.**: Intersection volume between hand mesh and object mesh (cm^3)
- **Chamfer**: Chamfer distance between the reconstructed and ground-truth objects in normalized units.
- **FID**: Fréchet Inception Distance for pose distribution matching
- **APD**: Average pairwise geodesic distance measuring generation diversity (degrees)

Generation uses {*FID*, *APD*, *dNN*}; denoising {Δ_q, *dNN*, *m2m*}; completion {Δ_q, *dNN*, *APD*, *FID*}; recovery {*MPJPE*, *MPVPE*, *I.V.*}, and {*obj-CD*, *obj-F@*τ} when object geometry is optimized.

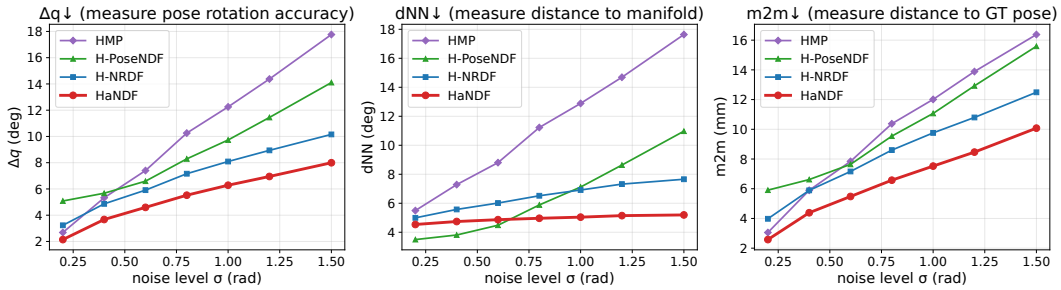

Figure 5: **Hand-pose denoising** on HOI4D under different noise levels ($\sigma \in [0.2, 1.5]$ rad). HaNDF (ours, red) achieves the best Δ_q and m2m at every noise level, with margins growing under heavier corruption.

5.1 Results on Downstream Tasks

Grasp pose generation. Given an object instance represented by its canonical geometry $\mathcal{O}$, the goal is to generate hand poses $\boldsymbol{\theta}_h$ that plausibly grasp $\mathcal{O}$. A successful generator should produce hand poses that are (i) *plausible* as natural human grasps, (ii) *compatible* with the conditioning object's geometry, and (iii) *diverse*, covering the many ways a human would naturally grasp the same object.

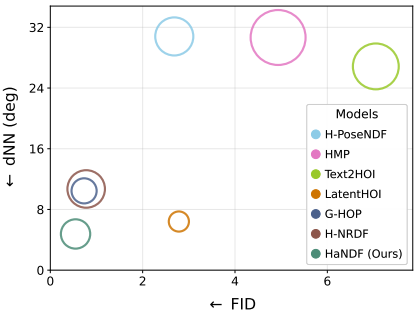

Figure 4: Pose generation on HOI4D: FID vs. dNN (lower is better); bubble size encodes APD.

As shown in Fig. 4, HaNDF lands in the bottom-left of the FID–dNN plane while maintaining a competitive bubble size, dominating both unconditional priors (H-NRDF, H-PoseNDF) and conditional generative baselines. As detailed in Tab. 7, our method achieves the best dNN (**4.77**°) and a highly competitive FID (0.54), indicating that our generated poses are both realistic and well-aligned with the data distribution. Although H-NRDF also reports low FID (0.77), its much higher dNN reveals limited per-sample realism. Compared to LatentHOI, HaNDF attains substantially higher diversity (APD 12.73° vs. 6.01°) while also improving FID and dNN, demonstrating a better balance between fidelity and diversity. We provide more pose generation results in Sec. D.1.

Hand pose denoising. We evaluate our hand pose denoising on the HOI4D test split with novel objects per category. For each clean test pose, we add noise at different levels σ, producing n=3000 noisy samples per σ; identical noisy inputs are fed to all methods to ensure an fair comparison. We report three metrics (lower-is-better for Δ_q, *dNN*, and *m2m*). All distance-field methods (HaNDF, H-NRDF, H-PoseNDF) denoise by gradient descent on the learned scalar field for the same iterations with the same step size. Except our method, none of the baselines can consider the object condition. We provide further results with varying object encoders in Sec. D.4, and more hand denosing results can be found in Sec. D.2.

As shown in Fig. 5 (supplemented by Tab. 8), HaNDF achieves the best Δ_q and m2m at every noise level, with margins that widen as σ grows. Most strikingly, HaNDF's *dNN* stays nearly constant across the full noise range ($4.54° \rightarrow 5.20°$). On dNN, HaNDF dominates from $\sigma \geq 0.8$ onward. At the smallest two σ, H-PoseNDF sits slightly closer to the manifold, but at the cost of much higher Δ_q and m2m. This is a mode-collapse signature, where outputs cluster near a few training centroids rather than tracking the input. Qualitative comparisons in Fig. 6 and Fig. 7 further validate these findings, showing that HaNDF consistently recovers plausible poses.

Hand completion. We now evaluate (object-conditioned) hand pose completion on the HOI4D test split under two masking protocols: **Missing Fingers**, where one of the 5 fingers (3 joints) is randomly masked per sample with the other 12 joints observed at GT; and **Missing Finger Parts**, where only the 5 finger-base joints are observed and the remaining 10 joints are masked. The masked joints are initialized with the same exponential noise, and n=3000 samples are drawn per task with identical inputs fed to all methods. Inference performs gradient descent on the learned scalar field, updating *only* the masked joints and keeping observed joints frozen. To isolate completion quality from the

Table 1: **Hand part completion** across two settings: missing fingers and partly missing fingers on HOI4D dataset.

Method	Missing Fingers				Missing Finger Parts			
	$\Delta_q \downarrow$	dNN↓	APD↑	FID↓	$\Delta_q \downarrow$	dNN↓	APD↑	FID↓
HMP [18]	24.37°	18.77°	**36.86°**	0.7407	16.60°	14.45°	**26.79°**	0.9448
H-PoseNDF	20.80°	16.40°	32.57°	1.0318	12.87°	8.83°	20.03°	0.7267
H-NRDF	16.60°	8.01°	24.48°	0.1558	8.47°	4.52°	12.17°	0.0453
HaNDF	**12.57°**	**5.44°**	20.79°	**0.0596**	**6.60°**	**3.57°**	12.23°	**0.0106**

trivially-correct observed joints. All metrics are computed on the masked-joint subset including Δ_q, *dNN*, *APD* , and *FID*.

As reported in Tab. 1, HaNDF achieves the best Δ_q, *dNN*, and *FID* on both completion tasks, cutting Δ_q by $\sim$25% and *FID* by 3–4$\times$ over the strongest baseline (H-NRDF), with even larger gaps against H-PoseNDF and HMP. The unconditional baselines follow a consistent ranking H-NRDF $>$ H-PoseNDF $>$ HMP. Notably, H-NRDF's gap to HaNDF widens on Missing Fingers (*dNN* 8.01° vs. 5.44°) compared to Missing Finger Parts (4.52° vs. 3.57°), highlighting that HaNDF's object conditioning compensates when more joints anchor the pose.

Table 2: **In-the-wild hand recovery** on HOI4D, OakInk, and DexYCB. For hand-focused, fair comparisons, the object ground truth is provided during inference. Lower is better.

Method	HOI4D [32]			OakInk [50]			DexYCB [8]		
	MPVPE↓	MPJPE↓	I.V.↓	MPVPE↓	MPJPE↓	I.V.↓	MPVPE↓	MPJPE↓	I.V.↓
IHoI [52]	19.42	20.12	41.44	14.75	15.17	1.66	29.85	30.73	1.26
EasyHOI [31]	16.64	17.06	13.28	13.17	13.48	0.93	29.64	30.08	2.61
EasyHOI + HaNDF	13.98	14.33	11.40	12.85	13.12	0.89	28.31	28.78	2.43
AlignSDF [12]	10.40	10.60	5.39	10.34	10.50	**0.82**	16.82	17.19	0.99
AlignSDF + HaNDF	**10.22**	**10.43**	**4.39**	**9.98**	**10.17**	0.87	**16.59**	**16.97**	**0.98**

Hand-object reconstruction. Single-image hand–object reconstruction remains a challenging task due to severe mutual occlusion, and serves as a natural downstream application of our prior. Since existing methods adopt heterogeneous object reconstruction pipelines (e.g. EasyHOI [31] relies on LISA + InstantMesh, while AlignSDF [12] uses a trained implicit SDF), we provide the ground truth object geometry to all methods for a fair comparison and focus the evaluation on hand reconstruction quality. We reserve optimizing for the object (shape recovery) in Tab. 3. We retrain and evaluate all methods on HOI4D, DexYCB, and OakInk under this unified protocol. To validate the effectiveness of our prior, we plug HaNDF into two representative baselines, EasyHOI and AlignSDF, as an object-conditioned hand prior that refines their hand results.

As shown quantitatively in Tab. 2 and qualitatively in Fig. 8, integrating HaNDF consistently improves two baselines across all three datasets. In particular, HaNDF (AlignSDF-based) achieves the best MPVPE, MPJPE, and intersection volume on most benchmarks, demonstrating that our prior provides a strong and generalizable regularization for hand reconstruction under occlusion. We provide more in-the-wild hand recovery results in Sec. D.3.

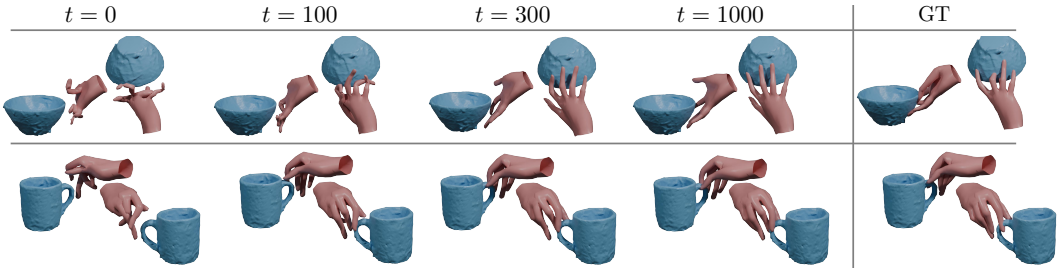

Figure 6: Qualitative pose denoising iterations for hand-object interaction on HOI4D. Starting from noisy initializations ($t = 0$), our model iteratively refines the hand poses. By $t = 1000$, the hand reaches a physically plausible configuration that closely matches the Ground Truth (GT).

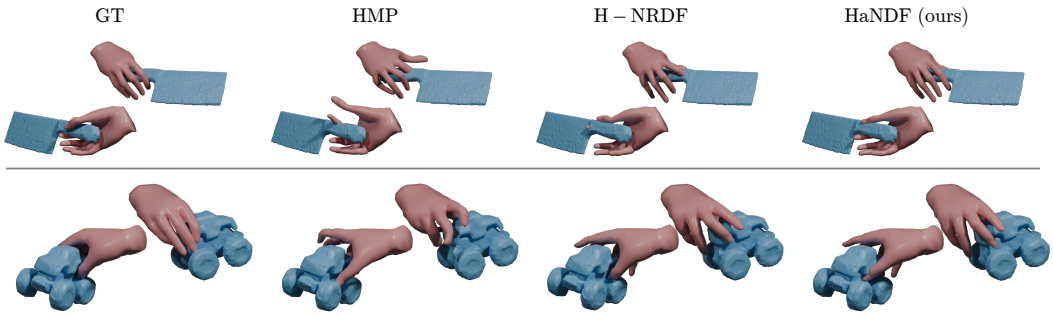

Figure 7: Qualitative comparison of pose denoising across different methods on HOI4D. We evaluate our method (HaNDF) against HMP and H-NRDF on a variety of object geometries. Ground Truth (GT) is provided for reference.

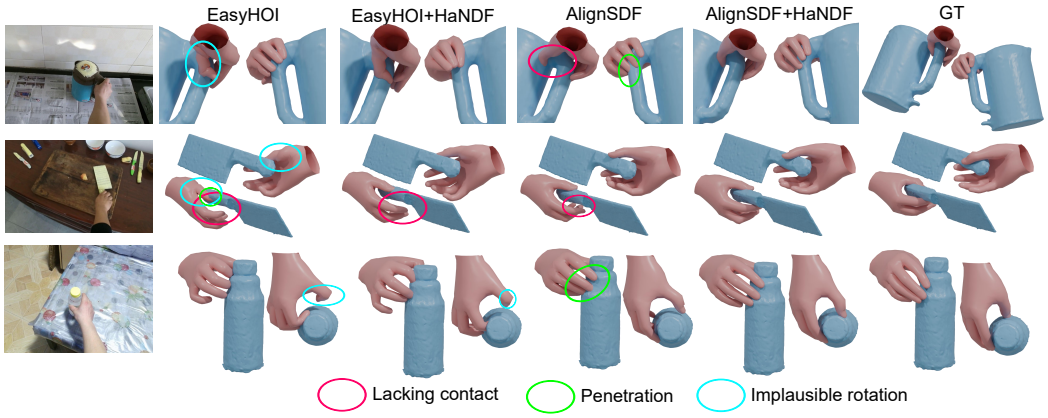

Figure 8: Qualitative results for in-the-wild hand recovery on HOI4D. Our HaNDF consistently improves the reconstruction quality of both EasyHOI and AlignSDF, alleviating common failure modes such as lacking contact, hand-object penetration, and implausible hand rotation.

Tab. 3 compares hand-object recovery on the HOI4D test set, using AlignSDF [12] as the baseline. Starting from AlignSDF's predictions, applying HaNDF as a denoiser substantially improves hand reconstruction quality, reducing Δq from 14.77 to 9.60, MPJPE from 12.08 to 11.74, and MPVPE from 11.90 to 11.52. This confirms that the learned hand prior, conditioned on the predicted object, effectively refines noisy hand outputs. We further refine the object mesh by retrieving the nearest neighbor in the DeepSDF latent space (with PCA), which not only improves object reconstruction (obj-F@2% from 0.1187 to 0.1568) but also further boosts hand quality (MPJPE: 11.74 $\rightarrow$ 11.50, MPVPE: 11.52 $\rightarrow$ 11.30). With additional object recovery (as shown in Alg. 2) with $K = 8$ optimization, we observe better hand performance, with slight worse object metrics. This experiment validates that better object reconstruction directly translates into better hand reconstruction, closing the loop between the two modalities.

Table 3: **Hand-object recovery.** We evaluate different hand priors built upon the hand–object reconstruction results from AlignSDF baseline [12] on the HOI4D dataset. We condition HaNDF on the object predicted by AlignSDF. We also refine the object by retrieving the nearest neighbor in the object latent space (with PCA), possibly with further object recovery by optimization with different K, which show advanced performance. CD refers to Chamfer Distance. The best and second-best results are highlighted in **bold** and underlined text, respectively.

Method	Obj retrieval	Obj recovery	$\Delta q \downarrow$	MPJPE↓	MPVPE↓	obj-CD↓	obj-F@1%↑	obj-F@2%↑
AlignSDF [12]	✗	N/A	14.77	12.08	11.90	0.0909	0.0296	0.1187
AlignSDF + HaNDF	✗	N/A	9.60	11.74	11.52	0.0909	0.0296	0.1187
	✓	$K = 1$	9.59	11.50	11.30	**0.0896**	**0.0449**	**0.1568**
	✓	$K = 8$	**9.57**	**11.45**	**11.25**	0.1083	0.0387	0.1360

Table 4: Ablation over backbone, conditioning, and field geometry on HOI4D.

Backbone / conditioning	Field geometry	Δ_q (°)	dNN (°)	m2m (mm)
MLP, unconditional	Euclidean	8.29	5.88	9.54
MLP, unconditional	Riemannian	7.16	6.52	8.60
MLP + object condition	Euclidean	8.15	5.12	9.06
MLP + object condition	Riemannian	5.98	**4.57**	7.14
Transformer, unconditional	Riemannian	6.10	5.75	7.40
Transformer + object condition	Euclidean	10.27	8.09	9.92
Transformer + object condition	Riemannian	**5.52**	4.96	**6.58**

5.2 Ablation Studies

Architecture We provide ablations on model architecture, conditioning, and the geometry of the neural field. As shown in Tab. 4, Riemannian geometry achieves a large performance gain, and object conditioning also improves performance for both MLPs and Transformers. With object conditioning and a Riemannian field, the MLP-based model also achieves comparable performance.

Object Conditioning To verify that HaNDF indeed exploits the object as a meaningful condition rather than treating it as a nuisance input, we compare three variants in Tab. 5. "Wrong condition" replaces the input object with a randomly sampled one from a different category, while "w/o object token" removes the object token from the model and retrains it with all other settings unchanged. Feeding a wrong object increases the joint-angle error, and removing the object token leads to an even larger drop in performance. These gaps confirm that HaNDF effectively leverages the object condition to guide hand denoising.

Table 5: Object-conditioning ablation on hand denoising with noise level σ=0.8 rad on HOI4D dataset.

	Δq (°) ↓	dNN (°) ↓	m2m ↓
Noisy input	11.74	12.43	11.13
HaNDF w/ wrong condition	5.80	5.25	7.05
HaNDF w/o object token	6.10	5.75	7.40
HaNDF	**5.52**	**4.96**	**6.58**

5.3 Runtime Analysis

To verify the runtime efficiency of the proposed HaNDF, we compare the time required to generate a single sample against LatentHOI [29] in Tab. 6. The experiment is done a single NVIDIA L40S GPU. Our proposed HaNDF (i) given the same iterations, can generate poses with substantially better FID and dNN scores faster than LatentHOI, and (ii) can match the LatentHOI metrics quicker, enabling less iterations and hence even quicker runtime. We provide more inference analysis in Sec. D.7

Table 6: Comparison of inference costs between the proposed HaNDF and LatentHOI [29] for object-conditioned generation on HOI4D. Runtime is reported per sample. †With 200 steps, the applied step size is doubled.

Method	# Steps	FID ↓	dNN ↓	Runtime (ms) ↓
LatentHOI [29]	1000	2.7843	6.41°	64.5
HaNDF	200†	0.5482	5.07°	**12**
HaNDF	500	0.5466	4.77°	29
HaNDF	1000	**0.5297**	**4.66°**	59

6 Conclusion

We presented HaNDF, an object-conditioned neural distance field for hand articulation on Riemannian manifolds. Coupled with manifold-aware optimization, the learned geometric prior supports denoising, completion, generation, and reconstruction. It can consistently improve robustness and physical plausibility across challenging benchmarks. More broadly, our results suggest that HaNDF provide a promising direction for learning structured interaction priors beyond Euclidean latent representations.

Limitations and future work. Our method relies on reasonably accurate object geometry and is not a true grasp prior as we do not model a hand-object joint NDF. Such an extension must account for the different geometries of hand rotations and object latents, along with their different noise scales and optimization step sizes, and may also need to incorporate hand–object contact dynamics. We will also extend our hand prior to bimanual interactions, as well as investigating more efficient training strategies. Finally, our model can be extended to the dynamic setting in a similar vein to NRMF [53]. More future work analysis are shown in Sec. H.

Acknowledgments and Disclosure of Funding

T. Birdal was supported by a UKRI Future Leaders Fellowship [grant number MR/Y018818/1]. The authors acknowledge support from the UK AI Research Resource (AIRR Isambard AI) through grant 0251-4584-0945-1 – TopoFound.

References

[1] Ayce Idil Aytekin, Helge Rhodin, Rishabh Dabral, and Christian Theobalt. Follow my hold: Hand-object interaction reconstruction through geometric guidance. *International Conference on 3D Vision*, 2026.

[2] Gary Becigneul and Octavian-Eugen Ganea. Riemannian adaptive optimization methods. In *International Conference on Learning Representations*, 2019.

[3] Heli Ben-Hamu, Omri Puny, Itai Gat, Brian Karrer, Uriel Singer, and Yaron Lipman. D-flow: Differentiating through flows for controlled generation. In *International Conference on Machine Learning*, pages 3462–3483. PMLR, 2024.

[4] Tolga Birdal and Umut Simsekli. Probabilistic permutation synchronization using the riemannian structure of the birkhoff polytope. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11105–11116, 2019.

[5] Tolga Birdal, Umut Simsekli, Mustafa Onur Eken, and Slobodan Ilic. Bayesian pose graph optimization via bingham distributions and tempered geodesic mcmc. *Advances in Neural Information Processing Systems*, 31, 2018.

[6] Nicolas Boumal. *An introduction to optimization on smooth manifolds*. Cambridge University Press, 2023.

[7] Junuk Cha, Jihyeon Kim, Jae Shin Yoon, and Seungryul Baek. Text2hoi: Text-guided 3d motion generation for hand-object interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1577–1585, 2024.

[8] Yu-Wei Chao, Wei Yang, Yu Xiang, Pavlo Molchanov, Ankur Handa, Jonathan Tremblay, Yashraj S Narang, Karl Van Wyk, Umar Iqbal, Stan Birchfield, et al. Dexycb: A benchmark for capturing hand grasping of objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9044–9053, 2021.

[9] Jiayi Chen, Yingda Yin, Tolga Birdal, Baoquan Chen, Leonidas J Guibas, and He Wang. Projective manifold gradient layer for deep rotation regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6646–6655, 2022.

[10] Ricky TQ Chen and Yaron Lipman. Flow matching on general geometries. In *The Twelfth International Conference on Learning Representations*.

[11] Zerui Chen, Shizhe Chen, Cordelia Schmid, and Ivan Laptev. gsdf: Geometry-driven signed distance functions for 3d hand-object reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12890–12900, 2023.

[12] Zerui Chen, Yana Hasson, Cordelia Schmid, and Ivan Laptev. Alignsdf: Pose-aligned signed distance fields for hand-object reconstruction. In *European Conference on Computer Vision*, pages 231–248. Springer, 2022.

[13] Zerui Chen, Rolandos Alexandros Potamias, Shizhe Chen, and Cordelia Schmid. Hort: Monocular hand-held objects reconstruction with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6046–6057, 2025.

[14] Julian Chibane, Gerard Pons-Moll, et al. Neural unsigned distance fields for implicit function learning. *Advances in Neural Information Processing Systems*, 33:21638–21652, 2020.

[15] In Cho, Youngbeom Yoo, Subin Jeon, and Seon Joo Kim. Representing 3d shapes with 64 latent vectors for 3d diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 28556–28566, 2025.

[16] Sammy Christen, Shreyas Hampali, Fadime Sener, Edoardo Remelli, Tomas Hodan, Eric Sauser, Shugao Ma, and Bugra Tekin. Diffh2o: Diffusion-based synthesis of hand-object interactions from textual descriptions. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–11, 2024.

[17] Hai Ci, Mingdong Wu, Wentao Zhu, Xiaoxuan Ma, Hao Dong, Fangwei Zhong, and Yizhou Wang. Gfpose: Learning 3d human pose prior with gradient fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4800–4810, 2023.

[18] Enes Duran, Muhammed Kocabas, Vasileios Choutas, Zicong Fan, and Michael J Black. Hmp: Hand motion priors for pose and shape estimation from video. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 6353–6363, 2024.

[19] Bowen Fu, Gu Wang, Chenyangguang Zhang, Yan Di, Ziqin Huang, Zhiying Leng, Fabian Manhardt, Xiangyang Ji, and Federico Tombari. D-sco: Dual-stream conditional diffusion for monocular hand-held object reconstruction. In *European Conference on Computer Vision*, pages 376–394. Springer, 2024.

[20] Yi Gu, Yukang Gao, Yangchen Zhou, Xingyu Chen, Yixiao Feng, Mingle Zhao, Yunyang Mo, Zhaorui Wang, Lixin Xu, and Renjing Xu. Hr: Pose distance fields for humanoid robots. *arXiv preprint arXiv:2602.04851*, 2026.

[21] Yana Hasson, Gul Varol, Dimitrios Tzionas, Igor Kalevatykh, Michael J Black, Ivan Laptev, and Cordelia Schmid. Learning joint reconstruction of hands and manipulated objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11807–11816, 2019.

[22] Yannan He, Garvita Tiwari, Tolga Birdal, Jan Eric Lenssen, and Gerard Pons-Moll. Nrdf: Neural riemannian distance fields for learning articulated pose priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1661–1671, 2024.

[23] Yannan He, Garvita Tiwari, Xiaohan Zhang, Pankaj Bora, Tolga Birdal, Jan Eric Lenssen, and Gerard Pons-Moll. Molingo: Motion-language alignment for text-to-human motion generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 38387–38398, 2026.

[24] Hanwen Jiang, Shaowei Liu, Jiashun Wang, and Xiaolong Wang. Hand-object contact consistency reasoning for human grasps generation. In *Proceedings of the International Conference on Computer Vision*, 2021.

[25] Angjoo Kanazawa, Michael J Black, David W Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7122–7131, 2018.

[26] Korrawe Karunratanakul, Jinlong Yang, Yan Zhang, Michael J Black, Krikamol Muandet, and Siyu Tang. Grasping field: Learning implicit representations for human grasps. In *2020 International Conference on 3D Vision (3DV)*, pages 333–344. IEEE, 2020.

[27] Muhammed Kocabas, Nikos Athanasiou, and Michael J Black. Vibe: Video inference for human body pose and shape estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5253–5263, 2020.

[28] Nilesh Kulkarni, Davis Rempe, Kyle Genova, Abhijit Kundu, Justin Johnson, David Fouhey, and Leonidas Guibas. Nifty: Neural object interaction fields for guided human motion synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 947–957, June 2024.

[29] Muchen Li, Sammy Christen, Chengde Wan, Yujun Cai, Renjie Liao, Leonid Sigal, and Shugao Ma. Latenthoi: On the generalizable hand object motion generation with latent hand diffusion. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 17416–17425, 2025.

[30] Hung Yu Ling, Fabio Zinno, George Cheng, and Michiel van de Panne. Character controllers using motion vaes. In *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH)*, volume 39. ACM, 2020.

[31] Yumeng Liu, Xiaoxiao Long, Zemin Yang, Yuan Liu, Marc Habermann, Christian Theobalt, Yuexin Ma, and Wenping Wang. Easyhoi: Unleashing the power of large models for reconstructing hand-object interactions in the wild. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 7037–7047, 2025.

[32] Yunze Liu, Yun Liu, Che Jiang, Kangbo Lyu, Weikang Wan, Hao Shen, Boqiang Liang, Zhoujie Fu, He Wang, and Li Yi. Hoi4d: A 4d egocentric dataset for category-level human-object interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21013–21022, 2022.

[33] Junzhe Lu, Jing Lin, Hongkun Dou, Ailing Zeng, Yue Deng, Xian Liu, Zhongang Cai, Lei Yang, Yulun Zhang, Haoqian Wang, and Ziwei Liu. Dposer-x: Diffusion model as robust 3d whole-body human pose prior. *arXiv preprint arXiv:2508.00599*, 2025.

[34] Naureen Mahmood, Nima Ghorbani, Nikolaus F Troje, Gerard Pons-Moll, and Michael J Black. Amass: Archive of motion capture as surface shapes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5442–5451, 2019.

[35] Jebastin Nadar, Simone Foti, and Tolga Birdal. Posed-flow: Versatile and guided flow matching model of human pose. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21165–21175, 2026.

[36] Mohamed Omran, Christoph Lassner, Gerard Pons-Moll, Peter Gehler, and Bernt Schiele. Neural body fitting: Unifying deep learning and model based human pose and shape estimation. In *2018 international conference on 3D vision (3DV)*, pages 484–494. IEEE, 2018.

[37] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 165–174, 2019.

[38] Georgios Pavlakos, Dandan Shan, Ilija Radosavovic, Angjoo Kanazawa, David Fouhey, and Jitendra Malik. Reconstructing hands in 3d with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9826–9836, 2024.

[39] Haozhe Qi, Chen Zhao, Mathieu Salzmann, and Alexander Mathis. Hoisdf: Constraining 3d hand-object pose estimation with global signed distance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10392–10402, 2024.

[40] Davis Rempe, Tolga Birdal, Aaron Hertzmann, Jimei Yang, Srinath Sridhar, and Leonidas J Guibas. Humor: 3d human motion model for robust pose estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11488–11499, 2021.

[41] Javier Romero, Dimitrios Tzionas, and Michael J. Black. Embodied hands: Modeling and capturing hands and bodies together. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 36(6), November 2017.

[42] Yonatan Shafir, Guy Tevet, Roy Kapon, and Amit H Bermano. Human motion diffusion as a generative prior. *arXiv preprint arXiv:2303.01418*, 2023.

[43] Mingyi Shi, Sebastian Starke, Yuting Ye, Taku Komura, and Jungdam Won. Phasemp: Robust 3d pose estimation via phase-conditioned human motion prior. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14725–14737, 2023.

[44] Guy Tevet, Sigal Raab, Brian Gordon, Yoni Shafir, Daniel Cohen-or, and Amit Haim Bermano. Human motion diffusion model. In *The Eleventh International Conference on Learning Representations*, 2023.

[45] Garvita Tiwari, Dimitrije Antić, Jan Eric Lenssen, Nikolaos Sarafianos, Tony Tung, and Gerard Pons-Moll. Pose-ndf: Modeling human pose manifolds with neural distance fields. In *European Conference on Computer Vision*, pages 572–589. Springer, 2022.

[46] Shibo Wang, Haonan He, Maria Parelli, Christoph Gebhardt, Zicong Fan, and Jie Song. Magichoi: Leveraging 3d priors for accurate hand-object reconstruction from short monocular video clips. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2025.

[47] Tony Danjun Wang, Tolga Birdal, Nassir Navab, and Lennart Bastian. Dispose: Projected polystochastic diffusion for self-supervised multi-view 3d human pose estimation. In *Proceedings of the 43rd International Conference on Machine Learning*, Proceedings of Machine Learning Research. PMLR, 2026.

[48] Thomas Weng, David Held, Franziska Meier, and Mustafa Mukadam. Neural grasp distance fields for robot manipulation. In *2023 IEEE International Conference on Robotics and Automation*, pages 1814–1821. IEEE, 2023.

[49] Jiayu Yang, Taizhang Shang, Weixuan Sun, Xibin Song, Ziang Cheng, Senbo Wang, Shenzhou Chen, Weizhe Liu, Hongdong Li, and Pan Ji. Pandora3d: A comprehensive framework for high-quality 3d shape and texture generation. *arXiv preprint arXiv:2502.14247*, 2025.

[50] Lixin Yang, Kailin Li, Xinyu Zhan, Fei Wu, Anran Xu, Liu Liu, and Cewu Lu. Oakink: A large-scale knowledge repository for understanding hand-object interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20953–20962, 2022.

[51] Yufei Ye, Abhinav Gupta, Kris Kitani, and Shubham Tulsiani. G-hop: Generative hand-object prior for interaction reconstruction and grasp synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1911–1920, 2024.

[52] Yufei Ye, Abhinav Gupta, and Shubham Tulsiani. What's in your hands? 3d reconstruction of generic objects in hands. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3895–3905, 2022.

[53] Zhengdi Yu, Simone Foti, Linguang Zhang, Amy Zhao, Cem Keskin, Stefanos Zafeiriou, and Tolga Birdal. Geometric neural distance fields for learning human motion priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2232–2242, 2026.

[54] Zhengdi Yu, Shaoli Huang, Yongkang Cheng, and Tolga Birdal. Signavatars: A large-scale 3d sign language holistic motion dataset and benchmark. In *European Conference on Computer Vision*, pages 1–19. Springer, 2024.

[55] Zhengdi Yu, Stefanos Zafeiriou, and Tolga Birdal. Dyn-hamr: Recovering 4d interacting hand motion from a dynamic camera. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27716–27726, 2025.

[56] Xinyu Zhan, Lixin Yang, Yifei Zhao, Kangrui Mao, Hanlin Xu, Zenan Lin, Kailin Li, and Cewu Lu. Oakink2: A dataset of bimanual hands-object manipulation in complex task completion. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 445–456. IEEE, 2024.

[57] Biao Zhang, Jiapeng Tang, Matthias Niessner, and Peter Wonka. 3dshape2vecset: A 3d shape representation for neural fields and generative diffusion models. *ACM Transactions On Graphics (TOG)*, 42(4):1–16, 2023.

[58] Yuhao Zhang, Gerard Pons-Moll, and Tolga Birdal. Samor: Motion modelling for articulated objects of any skeleton and topology. In *Advances in Neural Information Processing Systems*, 2026.

[59] Keyang Zhou, Bharat Lal Bhatnagar, Jan Eric Lenssen, and Gerard Pons-Moll. Toch: Spatio-temporal object-to-hand correspondence for motion refinement. In *European Conference on Computer Vision*, pages 1–19. Springer, 2022.

[60] Binghui Zuo, Zimeng Zhao, Wenqian Sun, Xiaohan Yuan, Zhipeng Yu, and Yangang Wang. Graspdiff: Grasping generation for hand-object interaction with multimodal guided diffusion. *IEEE Transactions on Visualization and Computer Graphics*, 31(9):5642–5654, 2024.

Appendix

A Riemannian Geometry

Let $\mathcal{M}$ be a smooth m-dimensional manifold embedded in an ambient Euclidean space $\mathcal{X} = \mathbb{R}^d$. For $\mathbf{x} \in \mathcal{M}$, the *tangent space* $T_{\mathbf{x}}\mathcal{M}$ is the set of all velocity vectors of smooth curves passing through $\mathbf{x}$:

$$T_{\mathbf{x}}\mathcal{M} = \{\dot{\gamma}(0) \mid \gamma : \mathbb{R} \to \mathcal{M},\ \gamma(0) = \mathbf{x},\ \gamma \text{ smooth}\}\,.$$

A *Riemannian metric* $\mathbf{G}$ assigns every $\mathbf{x} \in \mathcal{M}$ an inner product $\langle \mathbf{u}, \mathbf{v}\rangle_{\mathbf{x}} = \mathbf{u}^\top \mathbf{G}_{\mathbf{x}} \mathbf{v}$ for $\mathbf{u}, \mathbf{v} \in T_{\mathbf{x}}\mathcal{M}$, varying smoothly with $\mathbf{x}$. The pair $(\mathcal{M}, \mathbf{G})$ is a *Riemannian manifold*. For a smooth $f : \mathcal{M} \to \mathbb{R}$, the *Riemannian gradient* $\operatorname{grad} f(\mathbf{x}) \in T_{\mathbf{x}}\mathcal{M}$ is defined as the unique tangent vector satisfying

$$\mathrm{D}f(\mathbf{x})[\mathbf{v}] = \langle \mathbf{v}, \operatorname{grad} f(\mathbf{x})\rangle_{\mathbf{x}}, \qquad \forall \mathbf{v} \in T_{\mathbf{x}}\mathcal{M}. \tag{19}$$

For $\mathcal{M} \subseteq \mathcal{X}$ and f implemented for example by a neural network, automatic differentiation gives the Euclidean gradient $\nabla f(\mathbf{x}) \in \mathbb{R}^d$. The intrinsic gradient is obtained by projecting onto $T_{\mathbf{x}}\mathcal{M}$:

$$\operatorname{grad} f(\mathbf{x}) = \Pi_{\mathbf{x}}(\nabla f(\mathbf{x})), \tag{20}$$

where $\Pi_{\mathbf{x}} : \mathbb{R}^d \to T_{\mathbf{x}}\mathcal{M}$ is the orthogonal projection. This *egrad-to-rgrad* [6] conversion is the key enabler of our distance field operators. A *Riemannian gradient step* from $\mathbf{x}$ along a tangent vector $\boldsymbol{\xi} \in T_{\mathbf{x}}\mathcal{M}$ is performed by mapping $\boldsymbol{\xi}$ back to the manifold, ideally via the *exponential map* $\operatorname{Exp}_{\mathbf{x}} : T_{\mathbf{x}}\mathcal{M} \to \mathcal{M}$, yielding *Riemannian gradient descent* [6, 4] (RGD) of the form

$$\mathbf{x}_{t+1} = \operatorname{Exp}_{\mathbf{x}_t}\left(-\tau_t \operatorname{grad} f(\mathbf{x}_t)\right), \tag{21}$$

where $\tau_t > 0$ is a step size, which can often be adaptively computed, *e.g.*, via line search.

B The Rotation Group $\mathrm{SO}(3)$

For the sake of being self contained, we now present the space of 3D rotations as a Riemannian manifold, the *special orthogonal group*:

$$\mathrm{SO}(3) = \left\{\mathbf{R} \in \mathbb{R}^{3\times 3} \;\middle|\; \mathbf{R}^\top\mathbf{R} = \mathbf{I},\ \det(\mathbf{R}) = 1\right\}. \tag{22}$$

Its *Lie algebra* is the vector space of skew-symmetric matrices:

$$\mathfrak{so}(3) = \left\{\boldsymbol{\Omega} \in \mathbb{R}^{3\times 3} \;\middle|\; \boldsymbol{\Omega}^\top = -\boldsymbol{\Omega}\right\}. \tag{23}$$

For $\boldsymbol{\Omega} = (\omega_1, \omega_2, \omega_3)^\top \in \mathbb{R}^3$, define the hat operator $[\cdot]_\times$ so that $[\boldsymbol{\Omega}]_\times \mathbf{x} = \boldsymbol{\Omega} \times \mathbf{x}$ for all $\mathbf{x} \in \mathbb{R}^3$:

$$[\boldsymbol{\Omega}]_\times = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}, \tag{24}$$

The inverse map is denoted by $\operatorname{vee}(\cdot)$: $\operatorname{vee}([\boldsymbol{\Omega}]_\times) = \boldsymbol{\Omega}$. The tangent space of $\mathrm{SO}(3)$ at $\mathbf{R}$ is

$$T_{\mathbf{R}}\mathrm{SO}(3) = \{\mathbf{R}\boldsymbol{\Omega} \mid \boldsymbol{\Omega} \in \mathfrak{so}(3)\}\,. \tag{25}$$

Indeed, for a smooth curve $\mathbf{R}(t) \in \mathrm{SO}(3)$ with $\mathbf{R}(0) = \mathbf{R}$, differentiating the constraint $\mathbf{R}(t)^\top\mathbf{R}(t) = \mathbf{I}$ gives $\dot{\mathbf{R}}(0)^\top\mathbf{R} + \mathbf{R}^\top\dot{\mathbf{R}}(0) = \mathbf{0}$. Thus $\mathbf{R}^\top\dot{\mathbf{R}}(0)$ is skew-symmetric, and every tangent vector can be written as $\dot{\mathbf{R}} = \mathbf{R}\boldsymbol{\Omega}$ for some $\boldsymbol{\Omega} \in \mathfrak{so}(3)$. Using the Frobenius inner product $\langle \mathbf{A}, \mathbf{B}\rangle_F = \operatorname{Tr}(\mathbf{A}^\top\mathbf{B})$, the Riemannian gradient of $f : \mathrm{SO}(3) \to \mathbb{R}$ can be obtained from the Euclidean gradient $\nabla f(\mathbf{R})$ by

$$\operatorname{grad} f(\mathbf{R}) = \mathbf{R}\,\operatorname{skew}\big(\mathbf{R}^\top\nabla f(\mathbf{R})\big)\,, \quad \text{where} \quad \operatorname{skew}(\mathbf{A}) = \frac{1}{2}(\mathbf{A} - \mathbf{A}^\top). \tag{26}$$

This projection removes the normal component and keeps only the component tangent to $\mathrm{SO}(3)$. The *exponential map on* $\mathrm{SO}(3)$ is

$$\operatorname{Exp}_{\mathbf{R}}(\mathbf{R}\boldsymbol{\Omega}) = \mathbf{R}\exp(\boldsymbol{\Omega}), \tag{27}$$

where $\exp(\boldsymbol{\Omega})$ is the matrix exponential. If $\boldsymbol{\Omega} = [\boldsymbol{\Omega}]_\times$ and $\vartheta = \|\boldsymbol{\Omega}\|_2$, Rodrigues' formula gives

$$\exp(\boldsymbol{\Omega}) = \mathbf{I} + \frac{\sin\vartheta}{\vartheta}\boldsymbol{\Omega} + \frac{1-\cos\vartheta}{\vartheta^2}\boldsymbol{\Omega}^2. \tag{28}$$

The logarithmic map is the local inverse $\mathrm{Log}_{\mathbf{R}_1}(\mathbf{R}_2) = \mathbf{R}_1 \log(\mathbf{R}_1^\top \mathbf{R}_2)$, where $\log(\cdot)$ is the matrix logarithm. The geodesic distance between rotations is

$$d_{\mathrm{SO}(3)}(\mathbf{R}_1, \mathbf{R}_2) = \left\| \log(\mathbf{R}_1^\top \mathbf{R}_2) \right\|_{\mathfrak{so}(3)} = \cos^{-1}\left(\frac{\mathrm{Tr}(\mathbf{R}_1^\top \mathbf{R}_2) - 1}{2} \right). \tag{29}$$

These matrix expressions are useful conceptually, but for articulated hands we use *unit quaternions* because they provide a computationally convenient representation for gradient-based optimization.

C Implementation Details

C.1 Model Architecture

Our HaNDF applies a transformer encoder that processes hand and object tokens jointly. The architecture details are:

1. Hand token projection: Linear layer $\mathbb{R}^4 \to \mathbb{R}^{256}$ for each quaternion
2. Object token projection: Linear layer $\mathbb{R}^{64} \to \mathbb{R}^{256}$ for the single object token
3. [DIST] token: Learnable embedding $\in \mathbb{R}^{256}$ to aggregrate information
4. Positional encoding: Learnable positional embeddings for all tokens

As illustrated in Fig. 9, HaNDF adopts a multi-head transformer as its backbone by default. The 15 hand tokens, the object token, and a learnable distance token are concatenated and jointly processed by the backbone. The output distance token is then fed into a lightweight head to predict the final distance.

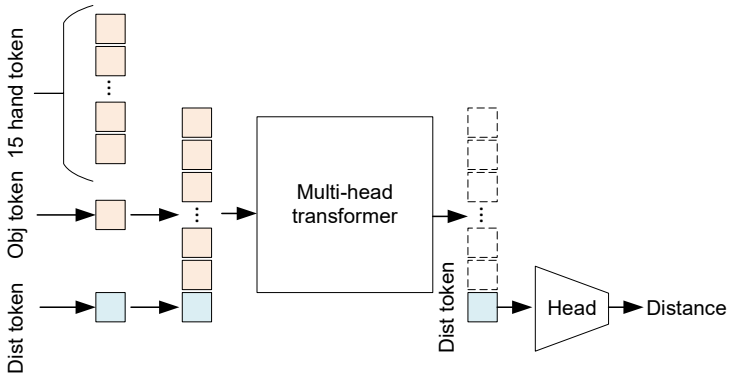

Figure 9: Diagram of HaNDF's transformer backbone, which combines hand, object, and distance tokens to predict the distance.

MLPs are also applicable as the backbone network for HaNDF, offering faster convergence at the cost of slightly degraded performance (as shown in Tab. 4). However, this performance gap does not necessarily preclude their applicability to downstream tasks, and in some cases MLP-based HaNDF can still achieve competitive results.

C.2 Training Algorithm

The algorithm to train HaNDF is as follow:

The training dataset normally stores data in the form of sequences, where we first get the object-hand pairs for interacting actions in the dataset to build the training set. Specifically, for each object $\mathcal{O}_j$, we first fit DeepSDF to obtain the object code z_j with 500 optimization steps, and we store the object code together with each hand pose under that object as a pair $(\boldsymbol{\theta}_{j,n}, \mathbf{z}_j, j)$, which serves as one training sample later. Note that for HOI4D and OakInk, we split the dataset based on objects, which means that the objects used during evaluation never appear during training. After constructing the training set, we sweep through the training samples to add perturbation noise based on an exponential distribution $\mathcal{P}$, and calculate the dNN between the noisy pose and all clean poses for that specific object to get the training label.

Algorithm 1 Training HaNDF

Require: Pose perturbation distribution $\mathcal{P}$, dataset of objects and associated hand poses $\mathcal{D} = \{(\mathcal{O}_j, \mathcal{D}_j)\}_{j=1}^{M}$, where $\mathcal{D}_j = \{\boldsymbol{\theta}_{j,n}\}_{n=1}^{N_j}$, initial parameters Φ of f_Φ
Ensure: Trained weights Φ

Initialize an empty training set $\mathcal{D}' \leftarrow \emptyset$
for each object $\mathcal{O}_j$ in $\mathcal{D}$ **do**
    Obtain object latent $\mathbf{z}_j$ by optimizing the DeepSDF autodecoder for 500 steps
    Add $(\boldsymbol{\theta}_{j,n}, \mathbf{z}_j, j)$ for all $n = 1, \ldots, N_j$ to $\mathcal{D}'$
end for
while not converged **do**
    for each epoch **do**
        Sample random subset $\mathcal{D}^O$ from $\mathcal{D}'$: $\mathcal{D}^O \subset \mathcal{D}'$
        Perturb all hand poses in $\mathcal{D}^O$ using the noise sampled from $\mathcal{P}$
        Update Φ by minimizing (using Adam)

$$\sum_{(\boldsymbol{\theta}', \mathbf{z}, j) \in \mathcal{D}^O} \left\| f_\Phi(\boldsymbol{\theta}' \mid \mathbf{z}) - \min_{\boldsymbol{\theta} \in \mathcal{D}_j} d_{\mathcal{M}_H}(\boldsymbol{\theta}', \boldsymbol{\theta}) \right\|$$

    end for
end while

C.3 Object Recovery

Alg. 2 summarizes the object recovery and hand refinement procedure. We first fit PCA to the mean-centred object codes in training set and retain the top-d principal directions to define the retrieval space. For each of the K candidates, we initialize an object latent code from $\mathcal{N}(\mathbf{0}, 0.01^2\mathbf{I})$ and optimize it using the fixed DeepSDF decoder. Specifically, we minimize the squared predicted SDF values at the input object surface points, whose target SDF values are zero. We then project the fitted code and the training codes into PCA space and retrieve the nearest training code as our retrieved object code.

Algorithm 2 Object recovery and hand refinement

Require: Initial hand pose $\boldsymbol{\theta}^0$, object surface points $\mathcal{O}^p$, training latent codebook $\mathcal{Z} = \{\mathbf{z}_j\}_{j=1}^{M}$, number of candidates K; PCA dimension d, hand refinement steps T, candidate evaluation steps T_c, coded-shape DeepSDF g_ψ

Compute the top-d principal directions $\mathbf{V}_d$ of the mean-centred codes in $\mathcal{Z}$
for $i = 1, \ldots, K$ **do**
    Initialize $\mathbf{z}_i^{(0)} \sim \mathcal{N}(\mathbf{0}, 0.01^2\mathbf{I})$
    Fit the object latent code from $\mathbf{z}_i^{(0)}$:
    $\hat{\mathbf{z}}_i \approx \arg\min_{\mathbf{z}} \frac{1}{|\mathcal{O}^p|} \sum_{\mathbf{o} \in \mathcal{O}^p} g_\psi(\mathbf{z}, \mathbf{o})^2$
    Retrieve the nearest training code in PCA space:
    $\tilde{\mathbf{z}}_i = \arg\min_{\mathbf{z} \in \mathcal{Z}} \left\| \mathbf{V}_d^\top (\hat{\mathbf{z}}_i - \mathbf{z}) \right\|_2$
    $\boldsymbol{\theta}_i^{(0)} \leftarrow \boldsymbol{\theta}_0$
    Compute $\boldsymbol{\theta}_i^{(T_c)}$ by iterating Eq. (HaNDFGrad) T_c times conditioned on $\tilde{\mathbf{z}}_i$
    $d_i^{(0)} \leftarrow f_\Phi(\boldsymbol{\theta}_i^{(0)} \mid \tilde{\mathbf{z}}_i)$
    $d_i^{(T_c)} \leftarrow f_\Phi(\boldsymbol{\theta}_i^{(T_c)} \mid \tilde{\mathbf{z}}_i)$
end for
$\mathbf{s} \leftarrow \mathrm{normalize}(\log(\mathbf{d}^{(0)}) - \mathrm{normalize}(\log(\mathbf{d}^{(T_C)}))$
$i^* \leftarrow \arg\min_{i \in \{1,\ldots,K\}} s_i$
Compute $\boldsymbol{\theta}_{i^*}^{(T)}$ by iterating Eq. (HaNDFGrad) T times conditioned on $\tilde{\mathbf{z}}_{i^*}$
return $\boldsymbol{\theta}_{i^*}^{(T)}, \tilde{\mathbf{z}}_{i^*}$

Starting from the initial hand pose, we apply T_c Riemannian updates conditioned on each retrieved object code. Our candidate selection is motivated by the intuition that a compatible object should require relatively little correction of the initial hand pose. To evaluate this, we record the predicted HaNDF distances before and after refinement. We take their logarithms and standardize the initial and final values separately across the K candidates. We then select the candidate object code with the smallest difference between its standardized initial and final distances. We finally refine hand with this selected object code. Quantitative results are shown in Tab. 3.

D Additional Results

D.1 Quantitative hand pose generation results

We evaluate hand pose generation on the HOI4D dataset using FID, APD, and dNN. The reults on Tab. 7 corresponds to visualization in Fig. 4 .

Table 7: Quantitative evaluation of pose generation on HOI4D dataset.

Model	FID ↓	APD ↑	dNN ↓
HMP [18]	4.9338	**44.72**°	30.66°
H-PoseNDF	2.685	21.48°	30.8°
H-NRDF	0.7769	20.86°	10.70°
G-HOP [51]	0.7316	9.34°	10.45°
LatentHOI [29]	2.7843	6.01°	6.41°
Text2HOI [7]	7.05	31.15°	26.85°
HaNDF (Ours)	**0.5466**	12.73°	**4.77**°

As shown in Tab. 7, HaNDF achieves the best FID (0.5466) and dNN (4.77°) scores, indicating that our generated poses most closely match the real data distribution while maintaining high fidelity. Although HMP attains a higher APD, this comes at the cost of substantially worse FID and dNN, suggesting that its diversity stems from generating implausible poses rather than capturing the true pose manifold. Overall, HaNDF strikes the best balance between fidelity and diversity among all competing methods.

D.2 More results on hand denoising

Quantitative evaluation. Tab. 8 provides the per-σ breakdown underlying Fig. 5. HaNDF achieves the best Δ_q and $m2m$ at all three noise levels, with widening margins as σ grows (Δ_q: $3.67°/5.52°/8.00°$; $m2m$: $4.38/6.58/10.08$), while its dNN remains nearly constant ($4.74° \rightarrow 5.20°$) and becomes the lowest from $\sigma \geq 0.8$. H-PoseNDF's marginally lower d_{NN} at $\sigma = 0.4$ ($3.81°$) is offset by substantially worse Δ_q ($5.67°$) and $m2m$ (6.62), evidencing mode collapse rather than better denoising. HMP trails all distance-field methods across every metric.

Table 8: Quantitative evaluation of hand denoising on HOI4D dataset across three representative noise levels. All methods receive identical noisy inputs.

	$\sigma = 0.4$ rad			$\sigma = 0.8$ rad			$\sigma = 1.5$ rad		
Model	dNN↓	$\Delta_q \downarrow$	m2m ↓	dNN↓	$\Delta_q \downarrow$	m2m ↓	dNN↓	$\Delta_q \downarrow$	m2m ↓
HMP [18]	7.29°	5.33°	5.90	11.23°	10.26°	10.37	17.64°	17.76°	16.38
H-PoseNDF	**3.81**°	5.67°	6.62	5.88°	8.29°	9.54	10.97°	14.10°	15.59
H-NRDF	5.57°	4.86°	5.89	6.52°	7.16°	8.60	7.66°	10.15°	12.49
HaNDF	4.74°	**3.67**°	**4.38**	**4.96**°	**5.52**°	**6.58**	**5.20**°	**8.00**°	**10.08**

In addition to the quantitative metrics presented in Tab. 8, we provide extended qualitative evaluations of our pose denoising method. Due to space constraints in the main manuscript, we present the remainder of our visual results here.

More qualitative results. Fig. 10 showcases further examples of the iterative denoising process. Starting from highly corrupted initial states, it is evident that over 1000 steps, HaNDF successfully

resolves penetrations and guides the hand toward physically plausible, tight grasps across diverse object topologies.

Furthermore, Fig. 11 provides additional baseline comparisons for the denoising task. Consistent with our main findings, our method produces hand-object configurations that align more accurately with the ground truth manifold, avoiding the artifacts observed in HMP and H-NRDF.

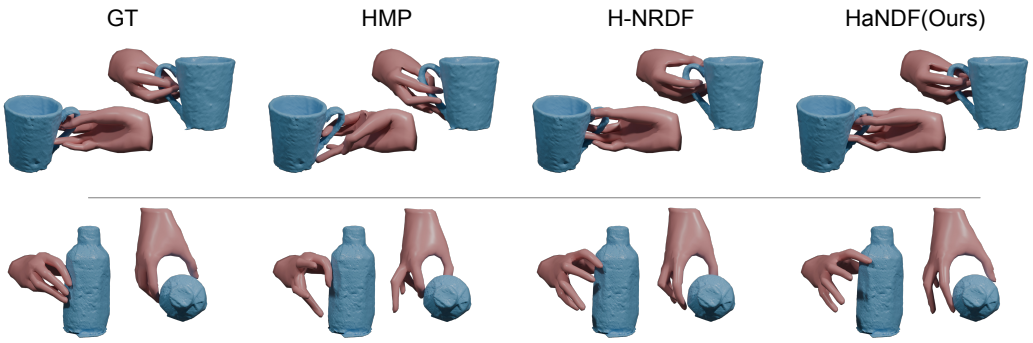

Figure 10: Extended qualitative comparison of pose denoising across different methods on HOI4D. We provide further comparison cases to supplement the results in the main paper. HaNDF (ours) is evaluated against HMP and H-NRDF, with Ground Truth (GT) provided for reference.

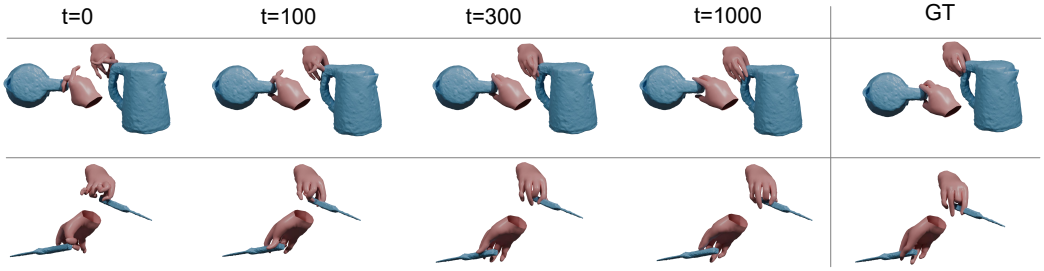

Figure 11: Additional qualitative pose denoising iterations for hand-object interaction on HOI4D. We show two further examples of the refinement steps from $t = 0$ to $t = 1000$, with Ground Truth (GT) provided for reference.

Robustness under controlled object-geometry perturbations.

We have also studied about robustness under controlled object-geometry perturbations. For each test object we sample 8,192 surface points on the mesh, apply a controlled corruption, re-fit the DeepSDF latent, and re-run denoising experiments. Entries are the cost of the corruption measured against the same fit on uncorrupted points. We evaluate robustness to three types of perturbations: point-wise noise at different noise levels σ_p, partial visibility, and point-cloud sparsity, and the results are shown in Tab. 9.

Conditioning is robust to corrupted object inputs. Point noise at $\sigma_p = 0.05$ costs 0.05° of Δ_{q_r} and occluding half the object costs 0.04°. Fits from occluded inputs land at cosine distance 0.63 from the clean fits, which is more than half the way to a different object. This shows the prior is insensitive to where the latent moves within a neighbourhood.

D.3 More results on in-the-wild hand recovery

Tab. 2 shows results under GT object code to focus on the hand performance. Here we further provide the EasyHOI-based end-to-end experiment by using EasyHOI-predicted objects to match to the nearest training object by shape, and then conditioning its latent on HaNDF to do the hand refinement.

Table 9: Robustness to object-input perturbations. We evaluate robustness to three types of perturbations: point-wise noise at different noise levels σ_p, partial visibility, and point-cloud sparsity. The results demonstrate good robustness across these perturbations.

Object input	Setting	$+\Delta_q$ (°)	+dNN (°)	+m2m (mm)
Point noise σ_p	0.01	+0.00	+0.00	+0.02
	0.03	+0.02	+0.06	+0.07
	0.05	+0.05	+0.06	+0.09
Partial visibility	25% occluded	+0.03	+0.05	+0.06
	50% occluded	+0.04	+0.07	+0.10
Point-cloud sparsity	75% subsample (control)	+0.00	+0.00	+0.01

Table 10: EasyHOI-based end-to-end experiment. We use EasyHOI-predicted objects to match to the nearest training object by shape, and then conditioning its latent on HaNDF to do the hand refinement.

Method	$\Delta_q(^\circ)$	MPVPE (mm)	MPJPE (mm)	IntVol (cm^3)
EasyHOI	20.82	16.64	17.06	13.28
EasyHOI + HaNDF	18.60	15.49	15.88	8.11
Improvement (%)	$+10.6$	$+6.9$	$+6.9$	$+38.9$

As shown in Tab. 10, the HaNDF performs reasonably well under end-to-end setting of in-the-wild recovery.

D.4 Comparison of object representations

In order to validate the efficiency of the applied DeepSDF-style auto-decoder for obtaining the latent code, we compare it with other object encoding methods, including COD-VAE [15] and Tencent-XR-3DGen [49] (based on 3DShape2VecSet [57]). The DeepSDF-style auto-decoder generates a single 64-dimensional vector, compared to 32×32 tokens for COD-VAE and 2048×64 for Tencent-XR-3DGen. Note that we retrain the HaNDF model under the same settings for each object encoder to keep the comparison fair.

Table 11: Ablation on object conditioning representations. We compare three pretrained object encoders as the source of object tokens fed into HaNDF on HOI4D.

Object encoder	$\Delta q \downarrow$	dNN $\downarrow$	m2m (mm) $\downarrow$
HaNDF+ COD-VAE [15]	6.11°	5.68°	7.46
HaNDF+ Tencent-XR-3DGen [49]	5.70°	5.17°	6.78
HaNDF+ DeepSDF-style auto-decoder	**5.52°**	**4.96°**	**6.58**

As shown in Tab. 11, the DeepSDF-style auto-decoder consistently outperforms both COD-VAE and Tencent-XR-3DGen, despite producing by far the most compact latent. Two factors account for this gap. First, the latter two are point-cloud-based encoders whose outputs are permutation-equivariant. Therefore, the point-cloud encodings carry no canonical ordering across instances, so the downstream transformer is hard to rely on positional cues to align semantically corresponding regions. Although we use the mean as the aggregation function for Tencent-XR-3DGen, it still underperforms compared to the applied DeepSDF-style auto-decoder. Second, the DeepSDF latent is optimized end-to-end to reconstruct the object's signed distance field, producing a compact, semantically aligned code in which geometrically similar shapes occupy nearby regions of the latent space. When the conditioning signal is too unstructured to be consumed effectively, HaNDF degenerates toward an unconditional hand prior, mirroring the "w/o object token" and "wrong condition" rows of Tab. 5.

D.5 Transformer architecture Ablation

We ablate three key design choices of our transformer architecture in Tab. 12 with depth and the granularity of the object condition. Our final configuration uses 4 layers, a single object token, and a dedicated distance token, which shows the better performance among different settings.

Table 12: Ablation on transformer architecture design choices on HOI4D dataset.

Variant	$\Delta_q \downarrow$	dNN $\downarrow$	Tokens(N)	Params	FLOPs (G)
2 layers	4.95°	6.08°	17	1.63	0.027
4 layers	5.52°	4.96°	17	3.21	0.054
8 layers	5.73°	5.07°	17	6.36	0.109
1 object tokens	**5.52**°	**4.96**°	17	3.21	0.054
64 object tokens	6.40°	6.09°	80	3.21	0.266

Depth. Increasing depth from 2 to 4 layers trades a small regression in Δ_q ($4.95^\circ \rightarrow 5.52^\circ$) for a substantial improvement in dNN ($6.08^\circ \rightarrow 4.96^\circ$), indicating that the shallower model fits per-joint quaternions slightly better but generalizes worse in the geodesic neighborhood sense. Pushing further to 8 layers degrades both metrics (5.73° / 5.07°) while doubling parameters and FLOPs. Given the relatively small input set (17 tokens) and the bounded complexity of the per-joint geodesic field, 4 layers offer the best accuracy/compute trade-off; deeper models appear to overfit without additional capacity benefit.

Object-token granularity. We compare encoding the 64-d DeepSDF object latent as a single condition token (a learned linear projection to the transformer width) against spreading it over 64 tokens (one per latent dimension). Despite the $5\times$ increase in attention FLOPs (0.054 G $\rightarrow$ 0.266 G), the 64-token variant is markedly worse (Δ_q: $5.52^\circ \rightarrow 6.40^\circ$). Splitting the 64-dimensional object code into 64 separate tokens breaks up the shape information, so the transformer must combine these tokens through attention. Using a single token keeps the full object code together and allows each hand-joint token to access it directly.

Results across random seeds. We repeat the comparison with and without object optimization over five random seeds. As shown in Tab. 13, object optimization reduces MPJPE by 0.236 ± 0.005 mm and MPVPE by 0.216 ± 0.005 mm, showing consistent improvements.

Table 13: Hand reconstruction results for AlignSDF + HaNDF over five random seeds on HOI4D. The gain is computed as the error without object optimization minus the error with object optimization.

Metric	Without obj. optim.	With obj. optim.	Gain
Δ_q (°)	9.5985 ± 0.0008	9.5920 ± 0.0011	0.0065 ± 0.0011
MPJPE	11.735 ± 0.002	11.499 ± 0.006	0.236 ± 0.005
MPVPE	11.516 ± 0.002	11.300 ± 0.005	0.216 ± 0.005

D.6 Ablation of PCA Dimension on Object Retrieval

During inference, the reconstructed object latent is projected into this PCA subspace, and nearest-neighbor retrieval is performed in the projected space. The retrieved latent is then used for the subsequent hand projection. Therefore, we also evaluate the effect of the PCA dimension used for nearest neighbor retrieval. As shown in Tab. 14, reconstruction performance improves as the dimension increases from 8 to 32 and remains similar beyond 32 dimensions. The default 32-dimensional setting achieves nearly identical performance to retrieval in the full latent space, indicating that dimensionality reduction has a negligible effect on retrieval quality.

One advantage for introducing PCA is scalability. As the object latent library grows, retrieval in the reduced space substantially lowers the search cost while preserving the optimization performance shown above. Since HaNDF is designed to retrieve the best candidate from an object library rather than generalize to unseen object categories, this provides an efficient implementation for large-scale retrieval.

Table 14: Effect of the PCA dimension on object retrieval on HOI4D. The 32-dimensional setting is used in Tab. 3. Note that 64 dimensions correspond to the full latent space.

PCA dimension K	Δ_q (°) ↓	MPJPE (mm) ↓	MPVPE (mm) ↓
2	9.68	11.73	11.53
8	9.74	11.70	11.50
16	9.63	11.63	11.42
32 (default)	**9.59**	**11.50**	**11.30**
48	9.60	11.53	11.33
64 (full latent)	9.59	11.52	11.32

D.7 Inference Efficiency

We evaluate the effect of the number of projection iterations on denoising accuracy and runtime. We use 3,000 poses with noise level $\sigma = 0.8$ and step size $\alpha = 0.02$, measuring runtime on a single NVIDIA L40S GPU at a batch size of 128. As shown in Tab. 15, 200 iterations already provide accuracy comparable to the default 500 iterations, while reducing the cost from 29 to 12 ms per pose. Increasing the iteration count beyond 500 provides no further improvement in these metrics.

Table 15: Denoising accuracy and runtime for different numbers of Riemannian projection iterations. The default setting uses 500 iterations.

Iterations	10	50	100	200	350	500	750	1000
Δ_q (°) ↓	10.18	6.97	5.86	**5.48**	5.49	5.52	5.56	5.57
d_{NN} (°) ↓	10.88	7.44	5.95	5.16	4.97	**4.96**	4.98	4.98
Time (ms/pose) ↓	**0.6**	3	6	12	21	29	44	59

D.8 Generalization to Unseen Object Categories

We conduct a category-shift experiment as an additional analysis to examine the behavior of the learned prior on object categories that are entirely absent from training. Specifically, we evaluate our HOI4D-trained model on three OakInk categories with no counterpart among the HOI4D categories, including sprayer, lotion pump, and camera. The test set includes 4 objects with 2,990 poses.

Table 16: Comparison of different models and settings.

Model / Setting	$\Delta_q(\hat{\circ})$ ↓	$d_{\mathrm{NN}}(\hat{\circ})$ ↓	$\mathrm{m2m(mm)}$ ↓
Noisy input	11.78	11.46	11.25
HOI4D-HaNDF (unseen category)	8.67	7.50	9.28
OakInk-HaNDF (in-category)	**5.01**	**3.77**	**6.58**

As shown in Tab. 16, the HOI4D-trained model improves all three metrics without adaptation, reducing Δ_q from $11.78°$ to $8.67°$, d_{NN} from $11.46°$ to $7.50°$, and $\mathrm{m2m}$ from 11.25 to 9.28 mm. These results suggest that the learned prior can improve hand poses even for object categories absent from training. However, a gap remains relative to the OakInk-trained model, indicating that category-specific training remains beneficial.

Fig. 12 shows qualitative examples of unseen object categories. In the successful cases, HaNDF corrects distorted finger configurations and brings the hand pose closer to the ground truth. The failure cases illustrate remaining limitations: fingers may remain overextended or fail to recover the reference grasp, even when some of the initial pose errors are corrected.

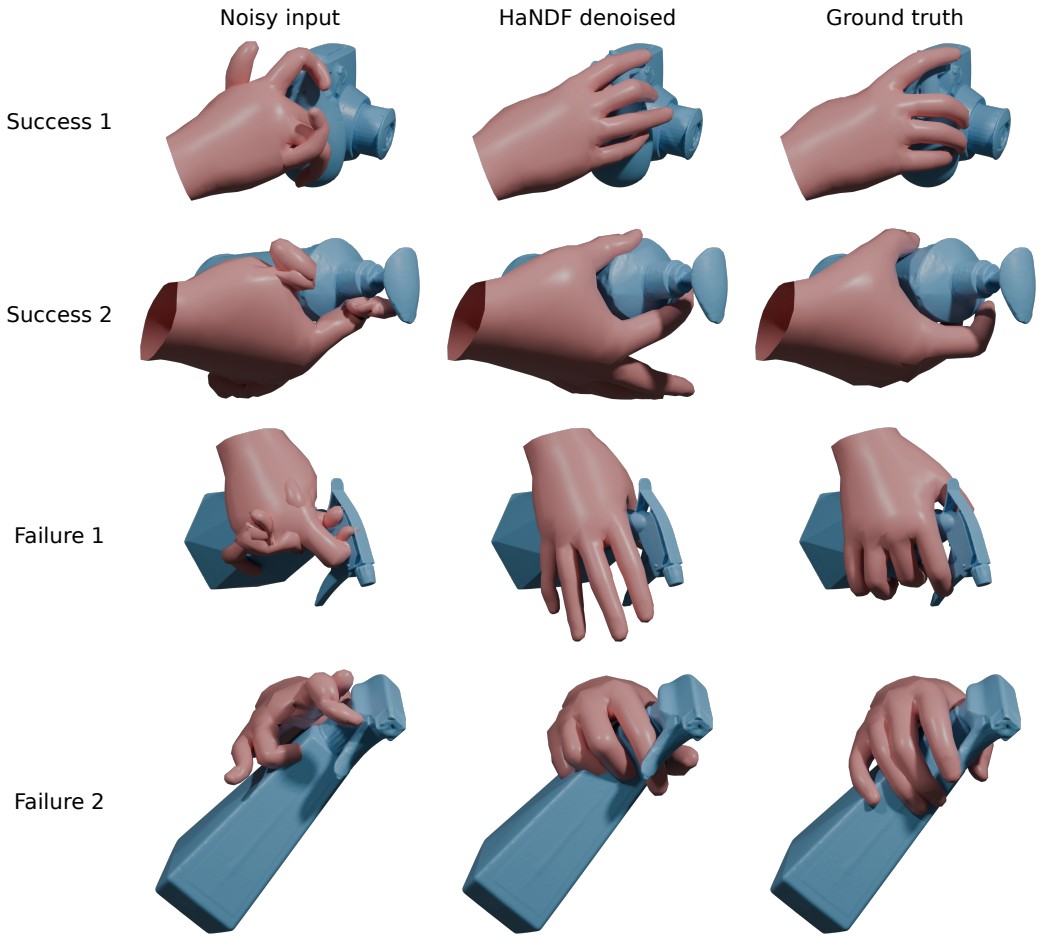

Figure 12: Generalization to Unseen Object Categories (by testing HOI4D trained HaNDF on OakInk unseen Categories).

E Baselines

Additional information on our baselines can be found below. We compare HaNDF against baselines spanning three categories:

1. Hand-domain adaptations of pose distance fields: PoseNDF [45] and NRDF [22] are distance-field priors originally proposed for full-body SMPL pose. Since no public hand counterpart exists, we re-implement both on the 15-joint MANO manifold and retrain them on hand-object datasets, denoted H-PoseNDF and H-NRDF. H-PoseNDF performs Euclidean gradient descent with per-step quaternion re-normalization, while H-NRDF follows the Riemannian formulation on $(\mathbb{S}^3)^{15}$. Both are unconditional (no object input), serving as the most direct ablations of HaNDF's object conditioning.
2. Hand / hand–object generative priors: HMP [18] is a learned unconditional hand-pose prior. G-HOP [51], LatentHOI [29], and Text2HOI [7] are recent hand-object generative models that condition on object geometry (and text for Text2HOI). These cover the current state of the arts in object-conditioned hand pose generation. For motion generation methods (Text2HOI [7] and LatentHOI [29]), we adapt them to the single-frame right-hand pose setting and retrain both models for a fair comparison.
3. Single-image hand–object reconstruction: For the reconstruction task we additionally compare with EasyHOI [31], AlignSDF [12], and IHoI [52]. To isolate the contribution of the hand prior from heterogeneous object pipelines, we feed ground-truth object geometry to all methods and evaluate hand reconstruction quality. HaNDF is plugged into EasyHOI and AlignSDF as an object-conditioned refinement prior on top of their pipelines.

F Datasets

Our experiments use three hand-object interaction datasets: HOI4D [32], DexYCB [8], and OakInk [50]. We also use the MANO hand model [41] to represent hand.

1. HOI4D [32] is a large-scale 4D egocentric dataset for category-level human-object interaction, containing approximately 2.4M RGB-D frames across 4,000+ sequences with 800 object instances, recorded at 1920×1080 with calibrated intrinsics. Each frame is densely annotated with MANO hand parameters, 6-DoF object poses, action labels, and 2D/3D segmentation. Following an object-based split, we focus on 6 rigid categories: instances N01–N40 (219 objects, 63K poses across 1,061 sequences) are used for training and the unseen instances N46–N50 (30 objects, 7.5K poses across 123 sequences) for testing, ensuring no object overlap between splits. We further apply contact-based filtering using the action annotations to retain only frames in which the hand physically interacts with the object.
2. DexYCB [8] is a large-scale benchmark of third-person hand grasping, comprising 1,000 grasping sequences performed by 10 subjects on 21 YCB objects. We adopt the standard `s0` protocol, yielding approximately 582K annotated frames. Each frame provides MANO pose and shape parameters, 6-DoF object poses, and pixel-level segmentation.
3. OakInk [50] is a large-scale hand-object interaction repository. We use its image-based subset, OakInk-Image, to validate our HaNDF. This subset contains ground-truth 3D MANO and object pose annotations. We use sequences from 7 object categories to build our training and test sets, with 53 objects (38K poses) for training and 14 unseen objects (10K poses) for testing.

G Evaluation Metrics

We now provide further details of the evaluation metrics.

1. dNN (distance to the nearest neighbor). For a predicted pose, dNN is the root-mean-square per-joint geodesic distance over the 15 finger joints between a pose and its nearest neighbor among the poses of the same object category in the training set. A small dNN indicates that the grasp lies close to the valid-pose manifold. dNN specifically penalizes off-manifold artifacts (such as twisted joints and implausible flexion).
2. Δ_q (quaternion geodesic distance to ground truth). Δ_q is the root-mean-square per-joint geodesic error between the predicted and ground-truth quaternions over the evaluated joint set (all 15 finger joints for denoising, and only the masked joints for completion).
3. FID (distribution fidelity). Following [22], we fit Gaussian distributions to the joint-angle features of the predicted and reference poses. FID is the Fréchet distance between these distributions, measuring differences in their means and covariances.
4. APD (average pairwise distance). For generation and completion, we draw $K = 10$ samples per condition and compute their average pairwise geodesic distance on $(\mathbb{S}^3)^{15}$. Higher APD indicates greater diversity among the generated grasps.
5. m2m (marker-to-marker distance). Errors in proximal joint rotations can shift from several downstream joints to the whole hand. To capture this effect, we also report m2m (marker-to-marker distance), the mean Euclidean distance between the predicted and ground-truth hand vertices.
6. MPJPE / MPVPE (3D reconstruction error). For single-image hand–object recovery we report the standard Mean Per-Joint and Mean Per-Vertex Position Errors between the predicted and ground-truth hand.
7. I.V. (interaction volume). I.V. (cm^3) is the volume of the intersection between the predicted hand mesh and the object mesh, computed by voxelizing both meshes at 1 mm resolution. It quantifies physical implausibility: a low MPJPE coupled with a high I.V. indicates a pose that is numerically close to GT but penetrates the object.
8. Object reconstruction metrics. For the object recovery experiment (Tab. 3), we independently scale each predicted and ground-truth mesh so that the longest half-side of its bounding box is one. We then report the symmetric Chamfer distance (obj-CD) and F-score (obj-F@τ) at thresholds $\tau \in \{0.01, 0.02\}$ in these normalized units.

H Future Work

As mentioned in Sec. 6, our future work will focus on joint hand-object distance field, motion prior extension and bimanual interaction.

As a preliminary study of the bimanual interactions, we provide initial results on OakInk2 [56], which annotates both left and right hand together with the subjects. We filter OakInk2 frames where both hands contact the same rigid object. It gives 30417 train poses over 30 objects and 3441 test poses over 5 held-out instances, split by object ID.

We refine the two-hand predictions of HaMeR [38] on the held-out objects using HaNDF to demonstrate its applicability to the bimanual setting. The results are shown in Tab. 17. To enable refinement of both hands, we increase the number of hand tokens in our transformer from 15 to 30 and retrain the prior.

Table 17: Effect of hand priors on bimanual hand refinement using the OakInk2 dataset. Results are reported for the left hand, right hand, and both hands. The best results are shown in bold.

Method	Left	Right	Both-hand mean
HaMeR	10.00 / 9.84	10.38 / 10.38	10.19 / 10.11
+ right-hand-only prior (15 joints)	–	**9.75 / 9.77**	–
+ bimanual prior (30 joints)	**9.13 / 8.98**	9.97 / 9.99	**9.55 / 9.49**

We observe that the bimanual results on right hand underperform the right-hand only model. We provide the follow explaination: Firstly, both the bimanual prior and the right-hand-only prior are trained from the same training set. However, for the right-hand-only model, we additionally mirror the left hand and use it as augmentation, effectively providing additional training examples for the right-hand prior. Secondly, adding additional hand increases the dimensionality of the optimization space, which may make projection and optimization more difficult. Thirdly, since our HaNDF is trained only on the right hand, so its zero-level set was specialized only through object conditioning.

In terms of the joint-field for hand-object interaction, a naive extension is denoising hand pose and object latent jointly. We found it is affected by two main phenomenon: The first is that the heterogeneity of manifolds: hand articulation sits on a compact product of spheres with a well-behaved geodesic structure, while DeepSDF latents live in an unbounded Euclidean space whose plausible region is defined only implicitly, by the distribution of the training latents. A single joint objective therefore has to commit to two different notions of distance at once. The second is a matter of scale. The amount of hand-pose noise and the amount of latent noise that correspond to a comparable degree of semantic corruption differ by orders of magnitude, so a shared step size is either too coarse for one field or too timid for the other. We therefore regard a properly normalized joint field our next-step work.

Second, we envision that neural distance fields are applicable to cross-embodiment scenarios [58] where different skeletons are jointly represented. We also like to extend our work to factor in motion and dynamics similar to and beyond [23, 40, 53, 54], and operate in multi-view settings [47].

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state that HaNDF is an object-conditioned neural distance field for hand articulation, with applications to generation, denoising, completion, and reconstruction. These claims are supported by the method formulation and experiments

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper includes a limitations and future work paragraph in Sec. 6, noting that the method relies on reasonably accurate object geometry and does not yet model a full joint hand-object neural distance field.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: The paper does not present new formal theorems requiring proofs. The mathematical sections define the Riemannian geometry, quaternion manifold, geodesic distances, and optimization operators used by the method.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper describes the evaluated datasets, baselines, metrics, downstream tasks, optimization settings, architecture, and training configuration. Additional implementation details are provided in the appendix.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The implementation has been publicly released, along with instructions for environment setup and reproducing the main experimental results. All experiments use publicly available datasets.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper specifies dataset splits, baselines, evaluation metrics, downstream task protocols, optimization iterations and step sizes, model architecture, optimizer, learning rate, batch size, number of epochs, noise sampling, and hardware.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

Justification: The paper reports quantitative comparisons across datasets and tasks on large test sets with fixed evaluation protocols, and reports standard deviation for certain experiments. Yet, to keep the computation budget at a reasonable level, for a large set of experiments, the current manuscript does not include error bars, confidence intervals, standard deviations, or statistical significance tests.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The appendix reports that HaNDF training uses a single NVIDIA A100 GPU and takes approximately 8 hours.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work develops a hand-object interaction prior and evaluates it on established research datasets. The paper does not involve collection of new private data, deceptive practices, or deployment on human subjects.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.

- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: HaNDF can benefit applications that require reliable hand–object interaction understanding, including AR/VR interfaces, assistive robotics, dexterous manipulation, and 3D perception under occlusion. By encouraging physically plausible hand poses conditioned on object geometry, the method may reduce reconstruction artifacts such as implausible joint configurations, missing contact, or hand–object penetration. Potential negative impacts arise if accurate hand–object reconstruction is used for privacy-sensitive behavioral tracking or surveillance without consent. The model may also inherit biases from existing HOI datasets, including limited coverage of hand shapes, grasp styles, object categories, and camera viewpoints. Finally, although HaNDF improves plausibility, it is not a safety guarantee; deployment in robotic manipulation or other safety-critical settings would require additional validation, uncertainty estimation, and fail-safe mechanisms.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The method is a geometric prior for hand-object interaction and does not introduce a high-risk generative model, sensitive dataset, or deployed decision-making system. The paper should nevertheless clarify that the method is intended for research use and should not be deployed in safety-critical robotic manipulation without further validation.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.

- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper cites the three existing datasets used in the experiments, HOI4D, DexYCB, and OakInk, and includes an appendix paragraph documenting their licenses and non-commercial research use. It also documents the MANO license and the licenses of the relevant dataset toolkits, and states that restricted MANO assets are not redistributed.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The paper introduces a well documented new method, model checkpoint, and code package as part of the current submission. We provide relevant documentation, license information, and usage instructions alongside the assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not conduct new crowdsourcing experiments or new human-subject data collection. It evaluates on existing public hand-object interaction datasets.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper does not involve new human-subject experiments or newly collected participant data. It uses previously released datasets, so IRB information should be inherited from the original dataset papers where applicable.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method development, experiments, and analysis in this work do not use LLMs as an important, original, or non-standard component. Any LLM use, if any, was limited to writing, editing, or formatting assistance and did not affect the scientific methodology, results, or claims.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.