NRDF: Neural Riemannian Distance Fields for Learning Articulated Pose Priors

Yannan He [1,2] Garvita Tiwari [1,2,3] Tolga Birdal [4] Jan Eric Lenssen [3] Gerard Pons-Moll [1,2,3]

[1]University of Tübingen, Germany, [2] Tübingen AI Center, Germany,
[3]Max Planck Institute for Informatics, Saarland Informatics Campus, Germany
[4]Imperial College London, United Kingdom
https://virtualhumans.mpi-inf.mpg.de/nrdf

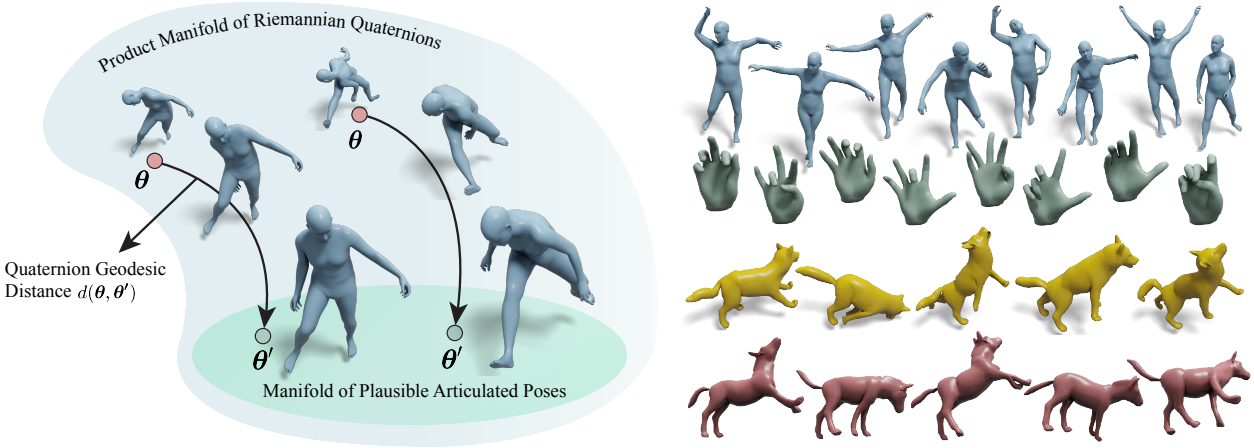


Figure 1. *Left*: We present **Neural Riemannian Distance Fields (NRDFs)**, a principled method to learn data-driven priors as subspace of high-dimensional Riemannian manifolds. *Right*: NRDFs can effectively model the pose of different articulated shapes. We present diverse samples generated using NRDFs trained on human, hand, and animal poses respectively.

Abstract

Faithfully modeling the space of articulations is a crucial task that allows recovery and generation of realistic poses, and remains a notorious challenge. To this end, we introduce Neural Riemannian Distance Fields (NRDFs), data-driven priors modeling the space of plausible articulations, represented as the zero-level-set of a neural field in a high-dimensional product-quaternion space. To train NRDFs only on positive examples, we introduce a new ***sampling algorithm****, ensuring that the geodesic distances follow a desired distribution, yielding a principled distance field learning paradigm. We then devise a* ***projection algorithm*** *to map any random pose onto the level-set by an* ***adaptive-step Riemannian optimizer****, adhering to the product manifold of joint rotations at all times. NRDFs can compute the Riemannian gradient via backpropagation and by mathematical analogy, are related to Riemannian flow matching, a recent generative model. We conduct a comprehensive evaluation of NRDF against other pose priors in various downstream tasks,* i.e.*, pose generation, image-based pose estimation, and solving inverse kinematics, highlighting NRDF's superior performance. Besides humans, NRDF's versatility extends to hand and animal poses, as it can effectively represent any articulation.*

1. Introduction

Pose and motion are ubiquitous yet very challenging and intriguing aspects of understanding articulated agents such as humans, animals or hands. Pose is intrinsic to the human experience, our interaction with each other and the environment. Understanding it is vital for applications in fields such as medicine, entertainment, AR/VR, *etc*. As a result, human pose understanding, generation, and acquisition have been extensively studied in the domain of computer vision and graphics.

Acquisition using IMUs [33], mocap markers [45], and scans [14] have accelerated the research direction by providing an enormous amount of data. These datasets are used to learn pose distributions, which are further used as priors in downstream tasks such as solving inverse-kinematics (IK), image HPS [51], motion denoising, *etc*. Previous work in this domain have used GMMs [13], VAEs [51] and

GANs [27] to model pose prior. However, these methods are either limited by Gaussian assumptions [13, 51] or risk suffering from instability of the training process [27].

In this work, we propose **Neural Riemannian Distance Fields (NRDFs)**, implicit, neural distance fields (NDFs) [24, 50] constructed on the space of plausible and realistic articulations. NRDFs are induced by the geodesic distance on the product manifold of quaternions and are trained to predict the Riemannian distance. In order to learn well-defined and detailed pose manifolds, we diligently study the role of training data distribution in learning distance field priors. To draw more samples near the surface with a gradual decrease for faraway regions [17], we introduce a **wrapped sampling algorithm on Riemannian manifolds** that allows explicit control over the resulting distance distribution. We show that heuristics developed in the past [63] sample points around the surface and often do not lead to desired distribution characteristics. The effect gets exacerbated in high-dimensional spaces like the product space of articulated bodies.

One of the key benefits of pose priors is the ability to map an arbitrary articulation onto a plausible one [63]. To this end, we introduce an **adaptive-step Riemannian gradient descent** algorithm, **RDFGrad**, in which the gradient obtained by a backward pass, scaled to the predicted distance, is used to update an articulated pose, respecting the product manifold of quaternions at all times. This is in stark contrast to Pose-NDF [63], which uses a Euclidean gradient descent to approximate the projection onto the articulation manifold. As a result, every projection step is followed by a re-projection onto joint rotations, resulting in slower convergence. Our manifold-aware formulation ensures the iterates to remain as articulated bodies. Our models bear similarities to Riemannian Flow Matching (RFM) [21, 44], the recent state-of-the-art generative models, as we explain. In fact, NRDF obtains the required gradients by backpropagation, whereas RFM is explicitly trained on them.

In summary, **our contributions are**:

- A principled framework for learning NDFs on Riemannian manifolds, with strong ties to flow matching
- A theoretically sound adaptive-step Riemannian gradient descent algorithm that leads to accelerated convergence in mapping poses onto the learned manifold
- A versatile framework for sampling training data, crucial for pose-manifold learning

The efficacy of NRDF is shown in a range of downstream tasks, such as pose generation, solving inverse kinematics (IK) problems, and pose estimation from images. We observe that the NRDF-based pose prior outperforms earlier works such as VPoser [51], GAN-S [27], GFPose [25], Pose-NDF [63] on aforementioned tasks, under pose distance metrics. We also conduct a user study about the perceptual quality of different pose distance metrics. As NRDF can be easily applied to any articulated shape, we also evaluate our model on hand poses and animal poses.

2. Related Work

We now review articulated (*e.g.* human) pose and motion priors crucial for understanding human pose from images [8, 13, 38, 42], videos [40, 60], IMUs [9, 33, 67] and scans [4, 14]. As we explore the connection to flow-matching models, we also review related literature therein.

Early pose priors. Initial works in modeling robust pose prior learn constraints for joint limits in Euler angles [29] or swing and twist representations [2, 6, 56]. However, these methods mainly rely on small-scale datasets and can still produce unrealistic poses due to unreal combinations of different joints. This was followed by more sophisticated models such as GMMs [13] or PCAs [49, 57, 66].

VAEs and GANs. Recent generative deep learning models have harnessed large-scale datasets to train VAEs [51–53, 71] and GANs [7, 27, 38] as pose/motion priors, either in a task-dependent [7, 38] or task-independent [27, 51] manner. For instance, [38] trains a GAN for $p(\boldsymbol{\theta}|I)$ for image-based pose estimation, HP-GAN [7] models $p(\boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1})$, representing the current pose given the previous one. HuMoR [53] proposes to learn a distribution of possible pose transitions in motion sequences using a conditional VAE. ACTOR [52] learns an action-conditioned VAE-Transformer prior. On the other hand, more task-independent models such as VPoser [51] learn a VAE using AMASS [45] dataset. However, because of Gaussian assumptions in the latent space, the model is biased towards generating mean poses, along with the risk of generating unrealistic poses from dead-regions of Gaussians [27, 63]. [27] learn a human pose prior using GANs, overcoming the limitations of Gaussians, but requires training a GAN, which is known to be unstable [55].

Probabilistic flow. More recently, pose and motion prior have also been developed using widely popular diffusion-based models [35, 58, 59]. GFPose learns score function (gradient of log-likelihood) of a task conditional distribution. Likewise, MDM [61] and MoFusion [26] model motion sequences conditioned on tasks through a diffusion process. Normalizing flows [39] have been applied in human pose-related tasks for a while, to address the ambiguous inverse 2D-to-3D problem [68], or recently to perform anomaly detection [34]. Flow Matching was also introduced to Riemannian manifolds [21]. We show that our method has strong ties to the flow matching principle and apply it to pose for the first time.

Distance fields. Closest to our work is Pose-NDF [63], which also models the manifold of plausible human poses using neural distance fields. Pose-NDF uses the learned distance field and its gradient to project arbitrary poses onto a manifold, using Euclidean gradient descent, where every step is followed by a re-projection onto the $SO(3)$. This results in slower convergence. In contrast, we leverage an

adaptive-step Riemannian gradient descent which ensures that the iterates always remain on $SO(3)^K$, yielding faster convergence. Moreover, Pose-NDF's training data generation is naively engineered, requiring a difficult per-task fine-tuning. We introduce a novel sampling method based on recent advances in scheduled optimal transport sampling [21], to create training data which results in robust learning without the need of manual, task-specific tuning.

3. Background

We first introduce the necessary preliminaries to define our Riemannian distance fields. Following [10, 11, 20], we define an m-dimensional *Riemannian manifold*, embedded in an ambient Euclidean space $\mathcal{X} = \mathbb{R}^d$ and endowed with a *Riemannian metric* $\mathbf{G} \triangleq (\mathbf{G}_\mathbf{x})_{\mathbf{x}\in\mathcal{M}}$ to be a smooth curved space $(\mathcal{M}, \mathbf{G})$. A vector $\mathbf{v} \in \mathcal{X}$ is said to be *tangent* to $\mathcal{M}$ at $\mathbf{x}$ *iff* there exists a smooth curve $\gamma : [0, 1] \to \mathcal{M}$ s.t. $\gamma(0) = \mathbf{x}$ and $\dot\gamma(0) = \mathbf{v}$. The velocities of all such curves through $\mathbf{x}$ form the *tangent space* $\mathcal{T}_\mathbf{x}\mathcal{M} = \{\dot\gamma(0) \,|\, \gamma : \mathbb{R} \to \mathcal{M}$ is smooth around 0 and $\gamma(0) = \mathbf{x}\}$, whose union is called the *tangent bundle*: $\mathcal{T}\mathcal{M} = \bigcup_\mathbf{x} \mathcal{T}_\mathbf{x}\mathcal{M} = \{(\mathbf{x}, \mathbf{v}) \mid \mathbf{x} \in \mathcal{M}, \mathbf{v} \in \mathcal{T}_\mathbf{x}\mathcal{M}\}$. The Riemannian metric $G(\cdot)$ equips each point $\mathbf{x}$ with an inner product in the tangent space $\mathcal{T}_\mathbf{x}\mathcal{M}$, $\langle \mathbf{u}, \mathbf{v}\rangle_x = \mathbf{u}^T\mathbf{G}_\mathbf{x}\mathbf{v}$. We will also work with a product of K manifolds, $\mathcal{M}_{1:K} := \mathcal{M}_1 \times \mathcal{M}_2 \times \cdots \times \mathcal{M}_K$, For identical manifolds, *i.e.* $\mathcal{M}_i \equiv \mathcal{M}_j$, we recover the *power manifold*, $\mathcal{M}^K := \mathcal{M}_{1:K}$, whose tangent bundle admits the *natural isomorphism*, $\mathcal{T}\mathcal{M}^K \simeq (\mathcal{T}\mathcal{M} \times \cdots \times \mathcal{T}\mathcal{M})$. We now define the operators required for our algorithm.

Definition 1 (Riemannian Gradient)**.** *For a smooth function $f : \mathcal{M} \to \mathbb{R}$ and $\forall(\mathbf{x}, \mathbf{v}) \in \mathcal{T}\mathcal{M}$, we define the* Riemannian gradient *of f as the unique vector field* $\operatorname{grad} f$ *satisfying [16]:*

$$\mathrm{D}f(\mathbf{x})[\mathbf{v}] = \langle \mathbf{v}, \operatorname{grad} f(\mathbf{x})\rangle_\mathbf{x} \tag{1}$$

where $\mathrm{D}f(\mathbf{x})[\mathbf{v}]$ is the derivation of f by $\mathbf{v}$. It can further be shown (see our supplementary) that an expression for $\operatorname{grad} f$ *can be obtained through the projection of the* Euclidean *gradient orthogonally onto the tangent space*

$$\operatorname{grad} f(\mathbf{x}) = \nabla f(\mathbf{x})_\parallel = \Pi_\mathbf{x}\big(\nabla f(\mathbf{x})\big). \tag{2}$$

where $\Pi_\mathbf{x} : \mathcal{X} \to \mathcal{T}_\mathbf{x}\mathcal{M} \subseteq \mathcal{X}$ is an orthogonal projector with respect to $\langle\cdot,\cdot\rangle_\mathbf{x}$.

In most packages such as ManOpt [64], Eq. (2) is known as the *egrad2rgrad*.

Definition 2 (Riemannian Optimization)**.** *We consider gradient descent to solve the problems of* $\min_{\mathbf{x}\in\mathcal{M}} f(\mathbf{x})$*. For a local minimizer or a* stationary point $\mathbf{x}^\star$ *of f, the Riemannian gradient vanishes* $\operatorname{grad} f(\mathbf{x}^\star) = 0$ *enabling a simple algorithm,* Riemannian Gradient Descent *(RGD):*

$$\mathbf{x}_{k+1} = R_{\mathbf{x}_k}(-\tau_k \operatorname{grad} f(\mathbf{x}_k)) \tag{3}$$

where τ_k is the step size at iteration k and $R_{\mathbf{x}_k}$ is the retraction *usually chosen related to the exponential map. Note that both RGD and its stochastic variant [15] are practically convergent [15, 16, 47, 65, 70]. Though, only in rare cases is τ_k analytically computable. Therefore, most minimizers use either Armijo or Wolfe line-search [1].*

Quaternions $\mathbb{H}_1$. A unit quaternion $\mathbf{q} \in \mathbb{H}_1$ represents a rotation using a 4D unit vector $[w := q_1, \mathbf{v} := (q_2, q_3, q_4)]$ double covering the non-Euclidean 3-sphere, *i.e.*, $\mathbf{q} \equiv -\mathbf{q}$ identify the same rotation. The inverse or *conjugate* of $\mathbf{q}$ is given by $\bar{\mathbf{q}} := \mathbf{q}^{-1} = (w, -\mathbf{v})$, whereas the non-commutative multiplication of two quaternions $\mathbf{q} = (q_1, \mathbf{v}_q)$ and $\mathbf{r} = (r_1, \mathbf{v}_r)$ is defined to be $\mathbf{q}\otimes\mathbf{r} := \mathbf{q}\mathbf{r} := (q_1 r_1 - \mathbf{v}_p\cdot\mathbf{v}_r,\ p_1\mathbf{v}_r + r_1\mathbf{v}_p + \mathbf{v}_p\times\mathbf{v}_r)$. Following [5, 12], we now briefly explain the Lie group structure of the quaternions essential for manifold optimization.

Definition 3 (Exponential map)**.** *The* exponential map $\operatorname{Exp}_\mathbf{q}(\cdot)$ *maps any vector in $\mathcal{T}_\mathbf{q}\mathbb{H}_1$ onto $\mathbb{H}_1$:*

$$\operatorname{Exp}_\mathbf{q}(\boldsymbol{\eta}) = \mathbf{q}\exp(\boldsymbol{\eta}) = \mathbf{q}\Big(\cos(\theta), \mathbf{v}\frac{\sin(\theta)}{\theta}\Big), \tag{4}$$

where $\boldsymbol{\eta} = (w, \mathbf{v}) \in \mathcal{T}_\mathbf{q}\mathbb{H}_1$ and $\theta = \|\mathbf{v}\|$.

Definition 4 (Logarithmic map)**.** *The inverse of* exp-map*,* $\operatorname{Log}_\mathbf{q}(\mathbf{p}) : \mathbb{H}_1 \to \mathcal{T}_\mathbf{q}\mathbb{H}_1$ *is* log-map *and defined as:*

$$\operatorname{Log}_\mathbf{q}(\mathbf{p}) = \log(\mathbf{q}^{-1}\mathbf{p}) = \Big(0, \frac{\mathbf{v}}{\|\mathbf{v}\|}\arccos(w)\Big), \tag{5}$$

this time with a slight abuse of notation $\mathbf{q}^{-1}\mathbf{p} = (w, \mathbf{v})$.

Definition 5 (Quaternion geodesic distance (d_q))**.** *Let us rephrase the Riemannian distance between two unit quaternions using the logarithmic map, whose norm is the length of the shortest geodesic path. Respecting the antipodality:*

$$d(\mathbf{q}_1, \mathbf{q}_2) = \begin{cases} \|\operatorname{Log}_{\mathbf{q}_1}(\mathbf{q}_2)\| \ \ = \arccos(w), & w \geq 0 \\ \|\operatorname{Log}_{\mathbf{q}_1}(-\mathbf{q}_2)\| = \arccos(-w), & w < 0 \end{cases}$$

where $\mathbf{q}_1^{-1}\mathbf{q}_2 = (w, \mathbf{v})$.

4. Neural Riemannian Distance Fields

We start by explaining *Riemannian Distance Fields* to model realistic articulated shapes in Sec. 4.1 and introduce our novel projection algorithm to map onto this space while adhering to the manifold of joint rotations. We then propose NRDF and a novel method for sampling articulated poses, to generate desired training data, in Sec. 4.2. We conclude this section by forming a link between recently-popularized flow matching models [21] and NRDF.

4.1. Modeling of Plausible Articulated Poses

We parameterize the pose of a 3D articulated body composed of K joints, $\boldsymbol{\theta} := \{\mathbf{q}_i \in \mathbb{H}_1\}_{i=1}^K$, on the power manifold of quaternions $\mathbb{H}_1^K = \mathbb{H}_1 \times \cdots \times \mathbb{H}_1$.

Definition 6 (Geometry of 3D articulated poses)**.** *$\mathbb{H}_1^K$ turns into a Riemannian manifold $(\mathbb{H}_1^K, \mathbf{G}^K)$ when endowed with the L_p* product metric *$d_{\mathbb{H}_1^K} : \mathbb{H}_1^K \times \mathbb{H}_1^K \to \mathbb{R}$:*

$$d_{\mathbb{H}_1^K}(\boldsymbol{\theta}, \boldsymbol{\theta}') = \|d(\mathbf{q}_1, \mathbf{q}_1'), d(\mathbf{q}_2, \mathbf{q}_2'), \ldots, d(\mathbf{q}_K, \mathbf{q}_K')\|_p,$$

where $\mathbf{q} \in \boldsymbol{\theta} \in \mathbb{H}_1^K$ and $\mathbf{q}' \in \boldsymbol{\theta}' \in \mathbb{H}_1^K$. In this work, we use $p = 1$. The natural isomorphism further allows us to write its exponential map $\mathrm{Exp}_{\boldsymbol{\theta}} : \mathcal{T}\mathbb{H}_1^K \to \mathbb{H}_1^K$ component-wise: $\mathrm{Exp}_{\boldsymbol{\theta}} = \left(\mathrm{Exp}_{\mathbf{q}_1}, \mathrm{Exp}_{\mathbf{q}_2}, \ldots, \mathrm{Exp}_{\mathbf{q}_K}\right)$. Akin to this, is the logarithmic map, $\mathrm{Log}_{\boldsymbol{\theta}}$. Since the tangent spaces and therefore $\Pi_{\boldsymbol{\theta}}$ are replicas, the gradient of a smooth function $f : \mathbb{H}_1^K \to \mathbb{R}$ w.r.t. $\boldsymbol{\theta}$ is also the Cartesian product of the individual gradients:

$$\mathrm{grad}_{\boldsymbol{\theta}} f(\boldsymbol{\theta}) = \left(\mathrm{grad}_{\mathbf{q}_1} f(\boldsymbol{\theta}), \ldots, \mathrm{grad}_{\mathbf{q}_K} f(\boldsymbol{\theta})\right). \quad (6)$$

Definition 7 (Riemannian Distance Fields (RDFs))**.** *Given the parameterization above, we model the manifold of* realistic and plausible articulations *(as defined by a dataset) on the zero level set of a model $f_\phi : \mathbb{H}_1^K \to \mathbb{R}^+$:*

$$\mathcal{S} = \{\boldsymbol{\theta} \in \mathbb{H}_1^K \mid f_\phi(\boldsymbol{\theta}) = 0\}, \quad (7)$$

such that the value of f represents the unsigned geodesic distance to the closest plausible pose on the manifold.

Proposition 1 (RDFGrad)**.** *Given $\mathcal{S}$ (hence f), we employ an adaptive-step Riemannian optimizer, to project any pose $\boldsymbol{\theta}_0$ onto the plausible poses:*

$$\boldsymbol{\theta}_{k+1} = \mathrm{Exp}_{\boldsymbol{\theta}_k}\left(-\alpha f(\boldsymbol{\theta}_k) \frac{\mathrm{grad} f(\boldsymbol{\theta}_k)}{\|\mathrm{grad} f(\boldsymbol{\theta}_k)\|}\right). \quad (8)$$

The details are given in Sec. 3. This procedure is in contrast to Pose-NDF [63], which uses a *projected* (Euclidean) gradient descent to approximate the projection onto the manifold. We therefore require the expression for projecting onto the tangent space of a quaternion, whose explicit form seems to be lacking in the literature. In what follows, we derive this operator.

Proposition 2 (Quaternion-egrad2rgrad)**.** *For the quaternion manifold, the projection and mapping onto the tangent space of the canonical unit quaternion $\mathbf{e} = \begin{bmatrix}1 & 0 & 0 & 0\end{bmatrix}^\top$ (egrad2rgrad in Eq. (2)) takes the form:*

$$\Pi_{\mathbf{q}}(\mathbf{v}) = \mathbf{P}\mathbf{v} - \frac{\mathbf{e}^\top \mathbf{P}\mathbf{v}}{1 + \mathbf{q}^\top \mathbf{e}}(\mathbf{q} + \mathbf{e}) \quad (9)$$

$$= \begin{bmatrix} 0 & 0 & 0 & 0 \\ -q_2/(1+q_1) & 1 & 0 & 0 \\ -q_3/(1+q_1) & 0 & 1 & 0 \\ -q_4/(1+q_1) & 0 & 0 & 1 \end{bmatrix} \mathbf{P}\mathbf{v}, \quad (10)$$

where $\mathbf{v} \in \mathbb{R}^4$ and $\mathbf{P}(\mathbf{q}) = \mathbf{I} - \mathbf{q}\mathbf{q}^\top$.

Sketch of the proof. The full proof uses the projection operator of $\mathcal{S}^3$ as well as the *parallel transport* of the quaternion manifold. We leave the full proof to our supplementary. □

4.2. Learning RDFs

We now describe how we construct $\mathcal{S}$, *i.e.*, learn f_ϕ.

Definition 8 (Neural RDFs (NRDFs))**.** *We model f using a combination of hierarchical network and an MLP decoder, similar to Pose-NDF [63]. Given a dataset $\mathcal{D} = \{\boldsymbol{\theta}_i\}_{1 \le i \le N}$ of articulated poses and a scheduled sampler for network inputs, the network is trained to predict the distance to the closest example from dataset $\mathcal{D}$:*

$$\phi^\star = \arg\min_\phi \sum_{i=1}^{N} \|f_\phi(\boldsymbol{\theta}_i) - \min_{\boldsymbol{\theta}' \in \mathcal{D}} d(\boldsymbol{\theta}_i, \boldsymbol{\theta}')\|. \quad (11)$$

We call $f_{\phi^\star}$, learned in this way, an NRDF*. We obtain $\mathrm{grad} f_{\phi^\star}(\boldsymbol{\theta})$ via backprop. followed by an* egrad2rgrad*.*

Sampling training data. While the positive examples, lying on $\mathcal{S}$ are provided, the model performance strongly depends upon the statistical distribution of training examples and including sensible negative samples ($d > 0$) is critical. This is also observed in the training of general neural distance fields for tasks like 3D shape reconstruction or completion [17, 23, 62]. To effectively capture intricate details of the pose manifold, it is essential to have an abundance of samples in proximity to the pose manifold $d < \epsilon$, gradually decreasing as we move away from it. This ensures that the network sees data points spanning the entire space, resulting in a well-behaved and continuous learned distance field.

Pose-NDF [63] samples a training pose as $\frac{\boldsymbol{\theta}+\epsilon}{\|\boldsymbol{\theta}+\epsilon\|_2}$, where $\boldsymbol{\theta} \sim \mathcal{D}$ and $\epsilon \sim \mathcal{N}(0, \sigma\mathbf{I}) \in \mathbb{R}^{4K}$. We observe the following: **(1)** This specific sampling technique leads to distances, which are roughly $\mathcal{X}$-distributed for large k before projection, as shown in Fig. 2a, as the distance is the sqrt-sum of squared Normal distributions with variance σ. This is contrary to the goal that the data should contain more samples close to the manifold. **(2)** Simply corrupting data samples by Euclidean noise does not expose explicit control over the distribution of generated distances, complicating the design of a schedule adhering to distance-related conditions.

In the following, we propose a framework for data sampling that allows for explicit control over generated distance distributions. As outlined in Alg. 1, given an arbitrary distribution $\mathcal{P}$ over $\mathbb{R}^+$ and an input pose $\boldsymbol{\theta}$, the algorithm first samples a distance $h \in \mathbb{R}$ and then generates an example, h apart from $\boldsymbol{\theta}$. It does so by sampling a direction $\mathbf{v} \in \mathcal{T}_{\boldsymbol{\theta}}\mathcal{M}^K$ independently from $h \in \mathcal{P}$, before finding the new pose via interpolation in tangent space. We can now show that the distances h sampled this way translate to the examples.

Proposition 3 (Distance preservation)**.** *Let $\mathcal{P}$ be a distribution over domain $[0, 1]$, $\boldsymbol{\theta} \in \mathcal{D}$ a data example, $\hat{\boldsymbol{\theta}} \in \mathbb{H}_1^K$ the output of Alg 1 with input $(\boldsymbol{\theta}, \mathcal{P})$, and $d = d(\boldsymbol{\theta}, \hat{\boldsymbol{\theta}})$. Then, for the distribution of resulting distances holds $p(d) = \mathcal{P}$.*

Sketch of the proof. The proof uses the distance preservation of logarithmic and exponential maps in the base. A full proof is given in the supplemental materials. □

Algorithm 1 Sampling in $SO(3)$ for articulated poses

Input: Data example $\boldsymbol{\theta}$, distribution $\mathcal{P}$
Output: A pair $(\hat{\boldsymbol{\theta}}, h)$, input to the network.
Sample distance from arbitrary $\mathcal{P}$:
1: $h \sim \mathcal{P}, h \in \mathbb{R}^+$
Sample direction $\mathbf{v}$ uniformly from unit sphere in $\mathcal{T}_{\boldsymbol{\theta}}\mathbb{H}_1^K$:
2: $\mathbf{v} \sim \mathcal{N}_{\mathcal{T}_{\boldsymbol{\theta}}\mathbb{H}_1^K}(\mathbf{0}, \mathbf{1}), \mathbf{v} \in \mathcal{T}_{\boldsymbol{\theta}}\mathbb{H}_1^K$
3: $\mathbf{v} \leftarrow \mathbf{v}/\|\mathbf{v}\|$
Interpolate in $\mathcal{T}_{\boldsymbol{\theta}}\mathbb{H}_1^K$ and map to $\mathbb{H}_1^K$:
4: $\hat{\boldsymbol{\theta}} \leftarrow \mathrm{Exp}_{\boldsymbol{\theta}}(h\mathbf{v})$

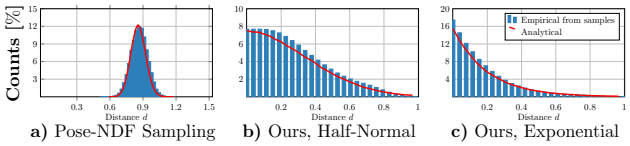
Given the introduced framework, we can induce a distribution $\mathcal{P}$, e.g. half-Gaussian, or exponential, as shown in Fig 2. Note that Prop. 3 holds for distances to the seed example $\mathbf{p}$, while we show the distribution of distances to the closest neighbor in $\mathcal{D}$ (c.f. Eq. (11)). Thus, we observe slight distribution shifts to the left.

Sampling diverse poses. To generate diverse pose samples on the manifold, we adopt an iterative procedure. We use Alg. 1 to produce an initial sample and then project it onto the zero level set via the proposed RDFGrad.

NRDF and Riemannian Flow Matching (RFM) [21]. RFM is a *simulation-free* method for learning continuous normalizing flows (CNFs) [22] on Riemannian manifolds, finding the *optimal transport* (OT) between a simple distribution and the data distribution. Interestingly, we can make a strong connection between our framework and flow matching. While flow matching predicts steps along OT trajectories towards the data manifold via feed-forward prediction, we find these steps as a gradient of our distance field via backpropagation. Our data generation procedure additionally ensures that (1) the $t \in [0, 1]$ of flow matching is a scaled variant of distance (by normalization of $\mathbf{v}$ in Alg. 1), and (2) we recompute nearest neighbors from $\mathcal{D}$ after sample generation. We provide a formal connection in the supplementals. In general, our distance field formulation has some advantages: instead of predicting the step towards the manifold via an autoencoder, we obtain it via backpropagation. This allows optimization in domains in which designing decoders is challenging. Also, we can utilize existing optimizers of deep learning frameworks for Lagrangian iterations by simply minimizing distance.

5. Experiments and Results

In this section, we evaluate the performance of NRDF on a range of downstream tasks and provide a comparison with baselines and prior work. NRDF incorporates three key components: an innovative sampling method for training data generation (Alg.1), Riemannian distance (d_q from Def 5), and a novel projection using RDFGrad. Ablation studies for each component are detailed in Sec. 5.1, alongside a comparison with Pose-NDF [63]. We also compare our model to score-based model [25] and RFM [21]-based work, emphasizing the strong mathematical connection between the latter and NRDF. We demonstrate the application of NRDF as a prior in downstream tasks, such as pose generation (Sec. 5.2), solving IK from sparse/partial observations (Sec. 5.3), and for human pose estimation from images (Sec. 5.3). We compare against previous pose priors such as VPoser [51], Pose-NDF [63], GFPose [25], GANS [27] and our own baselines. Since NRDF can be easily extended to any articulated shape, we show results on hand and animal poses in Sec. 5.4.



a) Pose-NDF Sampling **b)** Ours, Half-Normal **c)** Ours, Exponential

Figure 2. **Distance distributions and histograms from different sampling strategies**. **a)** Pose-NDF sampling generates a $\mathcal{X}$-like distribution for large k, which does not fit the needs of distance field learning. Our sampling schedule allows to control the distance distribution, e.g. to follow **b)** Half-Gaussian or **c)** Exponential distributions. The histograms show the distance to the closest example in $\mathcal{D}$, not the distance to the original example, resulting in a slight distribution shift to the left due to neighbor changes with increasing distance.

Evaluation metric. Previous research [19, 43] highlighted a significant gap in standard distance metrics based on joint locations, orientations, and user perception. We conducted a user study and identified that the most user-perceptive metric combines quaternion in the global frame and Euclidean marker-to-marker distance ($\Delta\mathbf{q}$ + m2m), followed by only marker-to-marker distance (m2m). These metrics are used for evaluation in our experiments, with user study details in the supplemental materials. To assess pose generation diversity, we utilize Average Pairwise Distance (APD) [3], and for realism evaluation, we employ Fréchet distance (FID) and a distance metric $d_{\mathrm{NN}} = \min_{\boldsymbol{\theta}' \in \mathcal{D}} d_q(\boldsymbol{\theta}, \boldsymbol{\theta}')$. FID gauges the similarity between the distributions of real and generated samples, while d_{NN} measures the distance between the generated pose and its nearest neighbor from the training data.

Baselines. Here, we present baselines inspired by flow matching and diffusion models. Specifically, we adopt a score-based human pose model, GFPose [25]. However, there are two key differences in the experimental setup: 1) GFPose is modeled using joint-location representation, and 2) the original model is trained on H3.6M [36] dataset. We implement two baselines based on GFPose: **GFPose-A** and **GFPose-Q**. GFPose-A, is trained using the AMASS dataset, in joint-location representation, and GFPose-Q, is trained using quaternion representations. We also introduce two RFM [21]-based baselines, namely **FM-Dis** and **FM-Grad**. FM-Grad represents the original RFM model trained for pose-denoising tasks with time conditioning, while FM-Dis closely aligns with our approach, where we predict the distance (without t-conditioning) and obtain the vector field

Method	Avg. Conv. Step↓	$\Delta\mathbf{q}$ +m2m ↓	m2m (cm) ↓
Pose-NDF [63]	40	0.349	25.04
GFPose-Q	\	0.359	24.43
Gradient Prediction w/o time	68	0.401	37.26
FM-Dis (w/o time)	33	0.230	18.74
FM-Grad (w/ time)	52	0.216	16.72
Ours (α=1.0)	**8**	**0.170**	**14.32**
Ours (α=0.5)	34	0.180	15.05
Ours (α=0.01)	34	0.201	16.59
Ours w/o RDFGrad	48	0.171	14.51
Ours w/o (RDFGrad, d_q)	100	0.179	15.15

Table 1. **Comparison with baselines and model ablations on pose denoising:** We evaluate ($\Delta\mathbf{q}$ +m2m) and m2m between denoised poses and their ground truth nearest neighbors. Our method achieves the best accuracy while converging faster, thanks to the novel training data generation, d_q and RDFGrad.

toward the manifold through backpropagation in contrast to the direct prediction of the vector field in FM-Grad. Models based on distance fields utilize gradients calculated through backpropagation to approach the manifold. Additionally, we implement a Gradient Prediction network, which directly predicts this gradient, and unlike diffusion, this approach is not time-conditioned. We provide more details in the supplementary. We apply the aforementioned baselines for pose denoising and generation. However, only FM-Dis is employed for optimization tasks, as others are formulated using the t-conditioned model, rendering them unsuitable for integration into the optimization pipeline.

5.1. Comparison with Baselines and Ablation Study

We first compare our model with prior works on the task of pose denoising in Tab. 1 (top) and evaluate using ($\Delta\mathbf{q}$ +m2m) and m2m. In contrast to the prior distance field-based model Pose-NDF [63], NRDF exhibits significantly lower error in the pose denoising task. We attribute this improvement to three key components of our model: training data generation, a Riemannian distance metric d_q, and RDFGrad-based projection. Specifically, unlike Pose-NDF, our training involves the sampling of more poses near the manifold, gradually decreasing as we move away from it. This method results in well-behaved training data, contributing to a continuous and more accurate learned manifold. Furthermore, the new RDFGrad-based projection ensures that the projection adheres to the manifold of poses, eliminating the need for re-projection as seen in Pose-NDF. Consequently, the convergence is faster, as indicated in Tab. 1. We note that GFPose-Q exhibits high error, primarily because denoising with GFPose-Q collapses to a mean pose. This suggests that training a diffusion/score-based model on quaternion representation poses challenges. Similar behavior is observed in FM-Dis, which tends to generate more common poses. The Gradient Prediction Network yields very high error, emphasizing the difficulty of training a gradient without time-conditioning. FM-Grad, which predicts flow with time conditioning, performs slightly better than the Gradient Prediction Network but still lags behind NRDF in terms of performance.

Method	FID ↓	APD↑ (in cm)	d_{NN} ↓ (in rad)
GMM [13]	$0.435^{\pm.017}$	$21.944^{\pm.102}$	$0.159^{\pm.001}$
VPoser [51]	$0.048^{\pm.002}$	$14.684^{\pm.138}$	$0.074^{\pm.000}$
GAN-S [27]	$0.201^{\pm.030}$	$10.914^{\pm.396}$	$0.098^{\pm.001}$
Pose-NDF [63]	$3.920^{\pm.034}$	$37.813^{\pm.085}$	$0.838^{\pm.001}$
GFPose-A	$1.246^{\pm.005}$	$13.876^{\pm.116}$	\
GFPose-Q	$1.624^{\pm.002}$	$6.773^{\pm.112}$	$0.159^{\pm.000}$
FM-Dis	$0.346^{\pm.007}$	$6.849^{\pm.199}$	$0.086^{\pm.001}$
Ours (α=0.01)	$0.636^{\pm.007}$	$23.116^{\pm.105}$	$0.177^{\pm.001}$

Table 2. **Pose generation**. We sample 20×500 poses. $\pm$ indicates the 95% confidence interval in sampling 20×500 poses.

Model ablation. We now perform ablation on each component of our model, including training data generation, Riemannian distance metric d_q, and RDFGrad-based projection. show results in Tab. 1(bottom). *Ours w/o RDFGrad* , yields similar error rates but exhibits slower convergence speed, indicating that RDFGrad-based projection adheres to the manifold and facilitates faster convergence. *Ours w/o* (RDFGrad, d_q) results in decreased accuracy, emphasizing that the new distance metric contributes to more accurate predictions. Finally, Pose-NDF is *Ours w/o* (RDFGrad, d_q) and w/o new training data, and we observe that the performance degrades significantly.

5.2. Diverse Pose Generation

We compare NRDF for pose generation with classical GMM [13, 48], VPoser [51], GAN-S [27], Pose-NDF [63], diffusion-based pose priors (GFPose-A, GFPose-Q), and RFM [21]-based model (FM-Dis), presenting the results in Tab. 2. Evaluating realism, VPoser shows the lowest FID and d_{NN}, indicating similarity to training samples, while Pose-NDF exhibits high FID and d_{NN}, suggesting more divergence from training samples but resulting in unrealistic poses. Meanwhile, FID and d_{NN} for NRDF are higher than VPoser and GAN-S, but lower than Pose-NDF. NRDF strikes a balance, producing diverse yet realistic poses.

Assessing pose diversity using APD reveals Pose-NDF with the highest values and VPoser with significantly lower scores. NRDF shows substantial APD, indicating more diversity than VPoser but less than Pose-NDF. However, given the large FID values, Pose-NDF tends to produce unrealistic poses, evident in Fig. 3 (a). VPoser, while less diverse, maintains realism. NRDF proves to be a good trade-off between diversity and realism, both visually and numerically. For the remaining priors, we note that they exhibit much lower APD, primarily generating mean poses.

5.3. Optimization-based Downstream Tasks

NRDF can be used as a pose prior term in optimization-based downstream tasks (Sec. 5.3) such as pose completion from partial observation or IK solver and pose estimation from images. For each task, we aim to find optimal SMPL parameters $(\boldsymbol{\theta}, \boldsymbol{\beta})$ that explain the observation. The optimization objective is formulated as Eq. (12), where $\mathcal{L}_{\text{data}}$ is the task-dependent data term, $\mathcal{L}_{\boldsymbol{\theta}}$ is the pose prior term, $\mathcal{L}_{\boldsymbol{\beta}}$

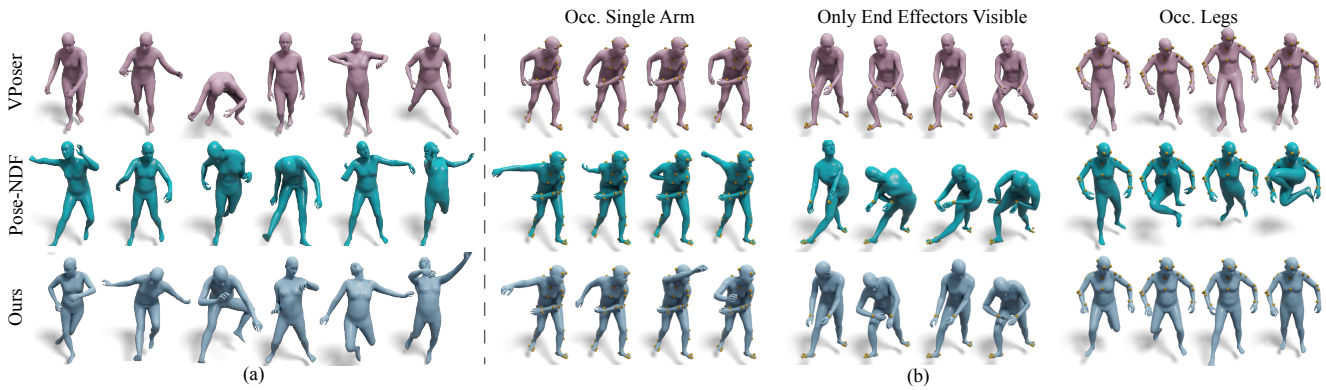


Figure 3. **(a) Pose generation: VPoser** generates realistic but somewhat limited diverse poses. **Pose-NDF** generates highly diverse poses but tends to yield unrealistic results (e.g., the third pose). **NRDF** demonstrates a balance between diverse and realistic poses. **(b) IK Solver from partial/sparse markers:** Given partial observation (yellow markers), we perform 3D pose completion. We observe that VPoser [51] based optimization generates realistic, yet fixed and less diverse poses. Pose-NDF [63] generates more diverse, but sometimes unrealistic poses, especially in case of very sparse observations. NRDF generates diverse and realistic poses in all setups.

is the shape prior term and $\mathcal{L}_\alpha$ represents any other regularizer, if needed. In our experiments we use $\mathcal{L}_{\boldsymbol{\beta}} = ||\boldsymbol{\beta}||^2$ [51]. For pose prior terms, VPoser uses $\mathcal{L}_\theta = ||z||_2^2$, where z is the latent vector of the VAE. Pose-NDF [63], FM-Dis and NRDF use $\mathcal{L}_\theta = f(\boldsymbol{\theta})$, where $f(\boldsymbol{\theta})$ is the distance value predicted from network. Details of the data and regularizer terms for each task are provided in respective sections.

$$\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\beta},\boldsymbol{\theta}}{\arg\min} \quad \mathcal{L}_{\text{data}} + \lambda_{\boldsymbol{\theta}}\mathcal{L}_{\boldsymbol{\theta}} + \lambda_{\boldsymbol{\beta}}\mathcal{L}_{\boldsymbol{\beta}} + \lambda_\alpha\mathcal{L}_\alpha, \quad (12)$$

IK solver from partial observations. Pose acquisition from dense sensors is expensive and tedious, while partial/sparse observation is underconstrained. This underscores the need for a fast and user-friendly Inverse Kinematics (IK) Solver for generating diverse and realistic complete poses from partial observations. To address this, we devise an experimental setup for 3D pose completion from partial observations. Our optimization process, based on Eq. (12), incorporates the $\mathcal{L}_{\text{data}}$ term defined by Eq. (13), where $M(\cdot)$ maps $(\beta, \boldsymbol{\theta})$ to SMPL mesh vertices, $\mathcal{J}$ maps the vertices to observed markers/joints, and $\boldsymbol{J}^{\text{obs}} \in \mathbb{R}^{|\text{joints}|\times 3}$ represents partial marker or joint observations.

$$\mathcal{L}_{\text{data}} = \sum_{i \in \text{joints}} ||(\mathcal{J}(M(\beta, \boldsymbol{\theta}))_j - \boldsymbol{J}_j^{\text{obs}})||_2 \quad (13)$$

We evaluate IK solver on three kinds of observations: 1) **Occluded Single Arm**, 2) **Only End Effectors Visible**, and 3) **Occluded Legs**. In Fig. 3 (b), we present qualitative results from our experiments, where multiple hypotheses are generated based on different initializations given a partial observation. Note that VPoser [51]'s default setting initializes the latent space with a zero vector without introducing noise. Thus we additionally fine-tune it with random initialization of latent space, denoting as **VPoser-Random**, where the initialization is sampled from a standard normal distribution $\mathcal{N}(0, \mathbf{I})$. Our findings show that VPoser-Random produces less diverse poses than distance field-based approaches due to the Gaussian assumption in VPoser's latent space. Additionally, Pose-NDF often generates unrealistic poses and is highly dependent on the initialization. In contrast, NRDF generates realistic and diverse poses.

Quantitative analysis in Tab. 3 includes evaluating pose diversity and realism using FID, APD, and d_{NN} metrics. Notably, VPoser tends to generate more common poses with slightly better FID and d_{NN} scores due to its generation of almost mean poses, consistently appearing realistic. Pose-NDF exhibits the highest APD, FID, and d_{NN} scores, indicating the generation of diverse yet unrealistic poses. These observations are consistent with the results shown in Fig. 3 (b). Our baseline, FM-Dis, performs poorly in terms of diversity and realism. Additional results on more experimental setups are provided in the supplementals.

Monocular 3D pose estimation from images. 3D pose estimation with neural networks is characterized by speed and robustness; however, it tends to lack accuracy due to the absence of feedback between the observation and prediction. To improve predictions, we propose refining the predictions through an optimization pipeline based on Eq. (12). We use the SotA pose-estimation model, SMPLer-X [18] for predictions and evaluate the optimization pipeline on the 3DPW [67] dataset. The data term in Eq. (12) is:

$$\mathcal{L}_{\text{data}} = \sum_{i \in \text{joints}} \gamma_i w_i \rho(\Pi_K(R_\theta(J(\boldsymbol{\beta}))) - \hat{J}_i) \quad (14)$$

where $\hat{J}_i$ are GT(or predicted) 2D-keypoints, R_θ transforms the joints along the kinematic tree according to the pose $\boldsymbol{\theta}$, Π_K is 3D-2D projection with intrinsic camera parameters, ρ is a robust Geman-McClure error [32], w_i are conf. factor of 2d keypoint prediction and γ_i is joint weight.

In Tab. 4, we compare NRDF-based optimization with other pose prior such as VPoser, Pose-NDF, FM-Dis, and also with NoPrior term. For quantitative evaluation, we measure PA-MPJPE, PA-PVE, and PCK@50mm [18, 67] on 3DPW dataset. Our experiments demonstrate that the optimization method based on NRDF consistently outper-

Method	Occ. Single Arm			Only End Effectors Visible			Occ. Legs		
	FID ↓	APD (in cm) ↑	d_{NN} (in rad) ↓	FID ↓	APD (in cm) ↑	d_{NN} (in rad) ↓	FID ↓	APD (in cm) ↑	d_{NN} (in rad) ↓
VPoser-Random	$1.148^{\pm.264}$	$3.218^{\pm.553}$	$0.069^{\pm.000}$	$0.769^{\pm.095}$	$6.706^{\pm.625}$	$0.068^{\pm.000}$	$0.650^{\pm.150}$	$9.399^{\pm1.368}$	$0.060^{\pm.004}$
Pose-NDF [63]	$1.281^{\pm.258}$	$15.294^{\pm1.927}$	$0.443^{\pm.001}$	$1.964^{\pm.125}$	$30.871^{\pm1.202}$	$0.643^{\pm.001}$	$3.043^{\pm.427}$	$30.291^{\pm1.987}$	$0.548^{\pm.001}$
FM-Dis	$1.341^{\pm.246}$	$4.490^{\pm1.293}$	$0.154^{\pm.001}$	$1.472^{\pm.252}$	$9.554^{\pm2.977}$	$0.153^{\pm.001}$	$1.030^{\pm.221}$	$7.950^{\pm2.773}$	$0.155^{\pm.001}$
Ours	$1.248^{\pm.341}$	$6.094^{\pm.003}$	$0.137^{\pm.000}$	$1.006^{\pm.144}$	$9.787^{\pm.040}$	$0.143^{\pm.000}$	$0.887^{\pm.170}$	$8.264^{\pm.007}$	$0.130^{\pm.000}$

Table 3. **Quantitative results for IK Solver from with partial/sparse markers.** We run all evaluations 20 times, ± indicates the 95% confidence interval. We evaluate under 3 settings: **Occ. Single Arm**, **Only End Effectors** (wrists and ankles) **Visible** and **Occ. Legs**. Our method generates more diverse poses than VPoser [51] for invisible body parts, while preserving more realistic poses (smaller distance to the manifold) than Pose-NDF [63] and FM-Dis.

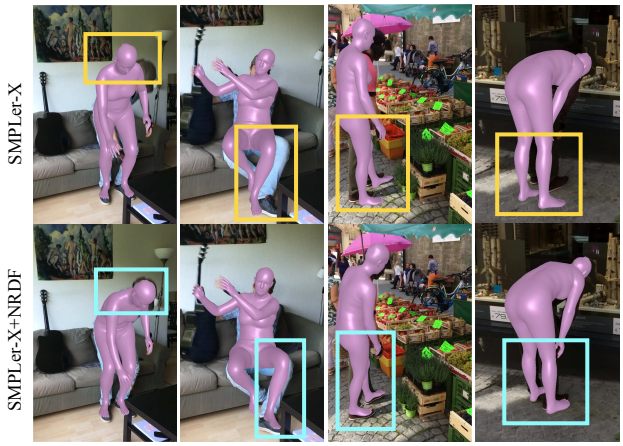


Figure 4. **3D pose and shape estimation from images:**(Top): Results from SMPLer-X [18], (Bottom): We refine the network prediction using NRDF based optimization pipeline. As highlighted, refined poses align better with the observation.

Method	PA-MPJPE↓ (in mm)	PA-PVE↓ (in mm)	PCK@50↑
SMPLer-X [18]	77.82	62.77	59.30
+ No prior	77.97	63.47	60.25
+ VPoser [51]	73.44	59.87	60.49
+ Pose-NDF [63]	70.19	55.30	62.06
+ FM-Dis	71.53	56.82	61.33
+ Ours	**68.47**	**53.29**	**66.17**

Table 4. **3D pose and shape estimation from images:** We take SMPLer-X [18] predictions and refine them using optimization pipeline. We compare the performance of different pose priors.

forms other pose priors. This highlights that the NRDF-based pose manifold is more detailed, leading to improved accuracy while preserving the realism of poses. We show qualitative results of SMPLer-X prediction and refined results using NRDF based optimization in Fig. 4 and provide more results in the supplementary material.

5.4. Extending NRDF to Other Articulated Bodies

As NRDF formulation is not limited to human poses, we use the same formulation to model manifolds of plausible hand and animal poses. For hands, we use MANO representation [54] and the DART dataset [31], covering 80K poses of the right hand, to learn a right-hand pose prior. For animals, we use SMALR [72, 73] representation and utilize the Animal3D dataset [69], covering 3K animal poses, to learn articulated animal pose priors. Specifically, we train two different priors for dogs and horses. We show diverse pose generation results in and Fig. 1 (right) and supplementals.

6. Conclusion and Discussion

We presented Neural Riemannian Distance Fields (NRDF), which are data-driven priors that model the space of plausible articulations. The pose prior is represented as the zero-level-set of a neural field in a high-dimensional product-quaternion space. Our model is trained to predict distance geodesic distance on the Riemannian pose manifold. We introduce crucial technical insights to effectively learn a well-behaved and detailed pose manifold. 1) We introduce a sampling framework on Riemannian manifold, that follows the desired distribution, 2) A Riemannian distance metric and 3) We develop a theoretically sound adaptive-step Riemannian gradient descent algorithm that accelerates the convergence in mapping poses onto the learned manifold. Furthermore, we establish connections with Riemannian flow matching [44] and introduce baselines based on RFM to demonstrate the advantages of NRDF. Our model demonstrates effectiveness in various applications, including pose generation, optimization-based Inverse Kinematics (IK) solving, and 3D pose estimation from images. We also show the versatility of our formulation by extending it to learning pose priors for hands and animals.

Limitations and future work. Since our approach is based on an iterative sampling scheme, it may slightly reduce efficiency for pose generation compared to directly mapping a random latent code to a pose. Rather than merely sampling the initial point, we could also *inject noise* during the projection, transforming it into a sequential geodesic MCMC sampler. This would be effective at generating a variety of random poses similar to a given initial pose. We could also model uncertainty over the manifold by describing it as a distribution over a family of implicit surfaces. We leave these promising avenues for future research.

Acknowledgments:. We thank RVH members, and reviewers for their feedback. The project was made possible by funding from the Carl Zeiss Foundation. This work is supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) - 409792180 (Emmy Noether Programme, project: Real Virtual Humans) and the German Federal Ministry of Education and Research (BMBF): Tübingen AI Center, FKZ: 01IS18039A. Gerard Pons-Moll is a member of the Machine Learning Cluster of Excellence, EXC number 2064/1 - Project number 390727645. This work was supported by the Engineering and Physical Sciences Research Council [grant number EP/X011364/1].

References

[1] P-A Absil and Kyle A Gallivan. Accelerated line-search and trust-region methods. *SIAM Journal on Numerical Analysis*, 47(2):997–1018, 2009. 3

[2] Ijaz Akhter and Michael J. Black. Pose-conditioned joint angle limits for 3D human pose reconstruction. In *CVPR*, 2015. 2

[3] Sadegh Aliakbarian, Fatemeh Sadat Saleh, Mathieu Salzmann, Lars Petersson, and Stephen Gould. A stochastic conditioning scheme for diverse human motion prediction. In *CVPR*, 2020. 5

[4] Thiemo Alldieck, Marcus Magnor, Bharat Lal Bhatnagar, Christian Theobalt, and Gerard Pons-Moll. Learning to reconstruct people in clothing from a single RGB camera. In *CVPR*, 2019. 2

[5] Jesus Angulo. Riemannian l p averaging on lie group of nonzero quaternions. *Advances in Applied Clifford Algebras*, 24(2):355–382, 2014. 3

[6] Paolo Baerlocher and Ronan Boulic. Parametrization and range of motion of the ball-and-socket joint. In *Proceedings of the IFIP TC5/WG5.10 DEFORM'2000 Workshop and AVATARS'2000 Workshop on Deformable Avatars*, 2000. 2

[7] Emad Barsoum, John Kender, and Zicheng Liu. HP-GAN: probabilistic 3D human motion prediction via gan. In *CVPR Workshops*, 2018. 2

[8] Bharat Lal Bhatnagar, Garvita Tiwari, Christian Theobalt, and Gerard Pons-Moll. Multi-garment net: Learning to dress 3D people from images. In *ICCV*, 2019. 2

[9] Bharat Lal Bhatnagar, Xianghui Xie, Ilya Petrov, Cristian Sminchisescu, Christian Theobalt, and Gerard Pons-Moll. BEHAVE: Dataset and method for tracking human object interactions. In *CVPR*, 2022. 2

[10] 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. 3

[11] 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. 3

[12] Tolga Birdal, Michael Arbel, Umut Simsekli, and Leonidas J Guibas. Synchronizing probability measures on rotations via optimal transport. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1569–1579, 2020. 3

[13] Federica Bogo, Angjoo Kanazawa, Christoph Lassner, Peter Gehler, Javier Romero, and Michael J Black. Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In *ECCV*, 2016. 1, 2, 6

[14] Federica Bogo, Javier Romero, Gerard Pons-Moll, and Michael J. Black. Dynamic FAUST: Registering human bodies in motion. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 2

[15] Silvere Bonnabel. Stochastic gradient descent on riemannian manifolds. *IEEE Transactions on Automatic Control*, 58(9): 2217–2229, 2013. 3

[16] Nicolas Boumal. An introduction to optimization on smooth manifolds. *Available online, May*, 2020. 3

[17] Ruojin Cai, Guandao Yang, Hadar Averbuch-Elor, Zekun Hao, Serge Belongie, Noah Snavely, and Bharath Hariharan. Learning gradient fields for shape generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 2, 4

[18] Zhongang Cai, Wanqi Yin, Ailing Zeng, Chen Wei, Qingping Sun, Yanjun Wang, Hui En Pang, Haiyi Mei, Mingyuan Zhang, Lei Zhang, et al. Smpler-x: Scaling up expressive human pose and shape estimation. *arXiv preprint arXiv:2309.17448*, 2023. 7, 8, 11

[19] Cheng Chen, Yueting Zhuang, Feiping Nie, Yi Yang, Fei Wu, and Jun Xiao. Learning a 3d human pose distance metric from geometric pose descriptor. *IEEE transactions on visualization and computer graphics*, 17, 2010. 5, 3

[20] 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. 3

[21] Ricky TQ Chen and Yaron Lipman. Riemannian flow matching on general geometries. *arXiv preprint arXiv:2302.03660*, 2023. 2, 3, 5, 6, 4

[22] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018. 5

[23] Julian Chibane, Thiemo Alldieck, and Gerard Pons-Moll. Implicit functions in feature space for 3D shape reconstruction and completion. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2020. 4

[24] Julian Chibane, Aymen Mir, and Gerard Pons-Moll. Neural unsigned distance fields for implicit function learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 2

[25] Hai Ci, Mingdong Wu, Wentao Zhu, Xiaoxuan Ma, Hao Dong, Fangwei Zhong, and Yizhou Wang. Gfpose: Learning 3d human pose prior with gradient fields. *arXiv preprint arXiv:2212.08641*, 2022. 2, 5, 6, 8

[26] Rishabh Dabral, Muhammad Hamza Mughal, Vladislav Golyanik, and Christian Theobalt. Mofusion: A framework for denoising-diffusion-based motion synthesis. In *Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[27] Andrey Davydov, Anastasia Remizova, Victor Constantin, Sina Honari, Mathieu Salzmann, and Pascal Fua. Adversarial parametric pose prior. In *CVPR*, 2022. 2, 5, 6, 8

[28] Manfredo Perdigao Do Carmo and J Flaherty Francis. *Riemannian geometry*. Springer, 1992. 2

[29] Morten Engell-Nørregård, Sarah Niebe, and Kenny Erleben. A joint-constraint model for human joints using signed distance-fields. *Multibody System Dynamics*, 28, 2012. 2

[30] Luca Falorsi, Pim de Haan, Tim R Davidson, and Patrick Forré. Reparameterizing distributions on lie groups. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 3244–3253. PMLR, 2019. 2

[31] Daiheng Gao, Yuliang Xiu, Kailin Li, Lixin Yang, Feng Wang, Peng Zhang, Bang Zhang, Cewu Lu, and Ping Tan.

DART: Articulated Hand Model with Diverse Accessories and Rich Textures. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. 8

[32] Donald Geman and Stuart Geman. Bayesian image analysis. In *Disordered systems and biological organization*, pages 301–319. Springer, 1986. 7

[33] Vladimir Guzov, Aymen Mir, Torsten Sattler, and Gerard Pons-Moll. Human POSEitioning System (HPS): 3D human pose estimation and self-localization in large scenes from body-mounted sensors. In *CVPR*, 2021. 1, 2

[34] Or Hirschorn and Shai Avidan. Normalizing flows for human pose anomaly detection, 2023. 2

[35] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. 2

[36] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1325–1339, 2014. 5

[37] Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547, 2019. 4

[38] Angjoo Kanazawa, Michael J. Black, David W. Jacobs, and Jitendra Malik. End-to-end recovery of human shape and pose. In *CVPR*, 2018. 2

[39] Ivan Kobyzev, Simon J.D. Prince, and Marcus A. Brubaker. Normalizing flows: An introduction and review of current methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):3964–3979, 2021. 2

[40] Muhammed Kocabas, Nikos Athanasiou, and Michael J. Black. Vibe: Video inference for human body pose and shape estimation. In *CVPR*, 2020. 2

[41] Max Kochurov, Rasul Karimov, and Serge Kozlukov. Geoopt: Riemannian optimization in pytorch, 2020. 6

[42] Nikos Kolotouros, Georgios Pavlakos, Michael J. Black, and Kostas Daniilidis. Learning to reconstruct 3D human pose and shape via model-fitting in the loop. In *ICCV*, 2019. 2

[43] Bjoern Krueger, Jochen Tautges, Andreas Weber, and Arno Zinke. Fast Local and Global Similarity Searches in Large Motion Capture Databases. In *Eurographics/ ACM SIGGRAPH Symposium on Computer Animation*. The Eurographics Association, 2010. 5, 3

[44] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2022. 2, 8

[45] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *ICCV*, 2019. 1, 2, 4

[46] Pertti Mattila. *Geometry of sets and measures in Euclidean spaces: fractals and rectifiability*. Number 44. Cambridge university press, 1999. 1

[47] Julien Munier. Steepest descent method on a riemannian manifold: the convex case. *Balkan Journal of Geometry & Its Applications*, 12(2), 2007. 3

[48] Mohamed Omran, Christop Lassner, Gerard Pons-Moll, Peter Gehler, and Bernt Schiele. Neural body fitting: Unifying deep learning and model based human pose and shape estimation. In *3DV*, 2018. 6

[49] Dirk Ormoneit, Hedvig Sidenbladh, Michael Black, and Trevor Hastie. Learning and tracking cyclic human motion. *Advances in Neural Information Processing Systems*, 13, 2000. 2

[50] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. DeepSDF: Learning continuous signed distance functions for shape representation. In *CVPR*, 2019. 2

[51] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3D hands, face, and body from a single image. In *CVPR*, 2019. 1, 2, 5, 6, 7, 8, 4, 9, 10

[52] Mathis Petrovich, Michael J. Black, and Gül Varol. Action-conditioned 3D human motion synthesis with transformer VAE. In *ICCV*, 2021. 2

[53] Davis Rempe, Tolga Birdal, Aaron Hertzmann, Jimei Yang, Srinath Sridhar, and Leonidas J. Guibas. HuMoR: 3D human motion model for robust pose estimation. In *ICCV*, 2021. 2

[54] 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), 2017. 8

[55] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, Xi Chen, and Xi Chen. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2016. 2

[56] Wei Shao and Victor Ng-Thow-Hing. A general joint component framework for realistic articulation in human characters. In *Proceedings of the 2003 symposium on Interactive 3D graphics*, pages 11–18, 2003. 2

[57] H. Sidenbladh, M. J. Black, , and D.J. Fleet. Stochastic tracking of 3D human figures using 2D image motion. In *ECCV*, 2000. 2

[58] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2019. Curran Associates Inc. 2

[59] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2021. 2

[60] Carsten Stoll, Juergen Gall, Edilson de Aguiar, Sebastian Thrun, and Christian Theobalt. Video-based reconstruction of animatable human characters. In *ACM SIGGRAPH Asia*, 2010. 2

[61] 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. 2

[62] Garvita Tiwari, Nikolaos Sarafianos, Tony Tung, and Gerard Pons-Moll. Neural-GIF: Neural generalized implicit functions for animating people in clothing. In *ICCV*, 2021. 4

[63] Garvita Tiwari, Dimitrije Antic, 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 (ECCV)*. Springer, 2022. 2, 4, 5, 6, 7, 8, 3, 9, 10

[64] J. Townsend, N. Koep, and S. Weichwald. PyManopt: a Python toolbox for optimization on manifolds using automatic differentiation. *Journal of Machine Learning Research*, 17(137):1–5, 2016. 3

[65] Nilesh Tripuraneni, Nicolas Flammarion, Francis Bach, and Michael I Jordan. Averaging stochastic gradient descent on riemannian manifolds. In *Conference On Learning Theory*, pages 650–687. PMLR, 2018. 3

[66] Raquel Urtasun, David Fleet, and Pascal Fua. 3D people tracking with Gaussian process dynamical models. In *CVPR*, 2006. 2

[67] Timo von Marcard, Roberto Henschel, Michael Black, Bodo Rosenhahn, and Gerard Pons-Moll. Recovering accurate 3d human pose in the wild using imus and a moving camera. In *European Conference on Computer Vision (ECCV)*, 2018. 2, 7

[68] Tom Wehrbein, Marco Rudolph, Bodo Rosenhahn, and Bastian Wandt. Probabilistic monocular 3d human pose estimation with normalizing flows. In *International Conference on Computer Vision (ICCV)*, 2021. 2

[69] Jiacong Xu, Yi Zhang, Jiawei Peng, Wufei Ma, Artur Jesslen, Pengliang Ji, Qixin Hu, Jiehua Zhang, Qihao Liu, Jiahao Wang, Wei Ji, Chen Wang, Xiaoding Yuan, Prakhar Kaushik, Guofeng Zhang, Jie Liu, Yushan Xie, Yawen Cui, Alan Yuille, and Adam Kortylewski. Animal3d: A comprehensive dataset of 3d animal pose and shape. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9099–9109, 2023. 8

[70] Hongyi Zhang and Suvrit Sra. First-order methods for geodesically convex optimization. In *Conference on Learning Theory*, pages 1617–1638. PMLR, 2016. 3

[71] Siwei Zhang, Haiyan Zhang, Federica Bogo, Marc Pollefeys, and Siyu Tang. Learning motion priors for 4D human body capture in 3D scenes. In *ICCV*, 2021. 2

[72] Silvia Zuffi, Angjoo Kanazawa, David W. Jacobs, and Michael J. Black. 3d menagerie: Modeling the 3d shape and pose of animals. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 8

[73] Silvia Zuffi, Angjoo Kanazawa, and Michael J Black. Lions and tigers and bears: Capturing non-rigid, 3d, articulated shape from images. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 3955–3963, 2018. 8