Unified Framework for Diffusion Generative Models in SO(3): Applications in Computer Vision and Astrophysics

Yesukhei Jagvaral[1,2], Francois Lanusse[3,4], Rachel Mandelbaum[1,2]

[1] McWilliams Center for Cosmology, Department of Physics, Carnegie Mellon University, Pittsburgh, PA 15213 USA
[2] NSF AI Planning Institute for Data-Driven Discovery in Physics, Carnegie Mellon University, Pittsburgh, PA 15213 USA
[3] AIM, CEA, CNRS, Universite Paris-Saclay, Universite Paris Cite, 91191 Gif-sur-Yvette, France
[4] Center for Computational Astrophysics, Flatiron Institute, New York, NY 10010, USA
yjagvara@andrew.cmu.edu

Abstract

Diffusion-based generative models represent the current state-of-the-art for image generation. However, standard diffusion models are based on Euclidean geometry and do not translate directly to manifold-valued data. In this work, we develop extensions of both score-based generative models (SGMs) and Denoising Diffusion Probabilistic Models (DDPMs) to the Lie group of 3D rotations, SO(3). SO(3) is of particular interest in many disciplines such as robotics, biochemistry and astronomy/cosmology science. Contrary to more general Riemannian manifolds, SO(3) admits a tractable solution to heat diffusion, and allows us to implement efficient training of diffusion models. We apply both SO(3) DDPMs and SGMs to synthetic densities on SO(3) and demonstrate state-of-the-art results. Additionally, we demonstrate the practicality of our model on pose estimation tasks and in predicting correlated galaxy orientations for astrophysics/cosmology.

1 Introduction

Deep generative models (DGM) are trained to learn the underlying data distribution and then generate new samples that match the empirical data. There are several classes of deep generative models, including Generative Adversarial Networks (Goodfellow et al. 2014), Variational Auto Encoders (Kingma and Welling 2013) and Normalizing Flows (Rezende and Mohamed 2015). Recently, a new class of DGMs based on Diffusion, such as Denoising Diffusion Probabilistic Models (DDPM) (Ho, Jain, and Abbeel 2020) and Score Matching with Langevin Dynamics (SMLD) , a subset of general score-based generative models (SGMs), (Song and Ermon 2019), have achieved state-of-the-art quality in generating images, molecules, audio and graphs[1] (Song et al. 2021). Unlike GANs, training diffusion models is usually very stable and straightforward, they do not suffer as much from mode collapse issues, and they can generate images of similar quality. In parallel with the success of these diffusion models, (Song et al. 2021) demonstrated that both SGMs and DDPMs can mathematically be understood as variants of the same process. In both cases, the data distribution is progressively perturbed by a noise diffusion process defined by a specific Stochastic Differential Equation (SDE), which can then be time-reversed to generate realistic data samples from initial noise samples.

While the success of diffusion models has mainly been driven by data with Euclidean geometry (e.g., images), there is great interest in extending these methods to manifold-valued data, which are ubiquitous in many scientific disciplines. Examples include high-energy physics (Brehmer and Cranmer 2020; Craven et al. 2022), astrophysics (Hemati et al. 2019), geoscience (Gaddes, Hooper, and Bagnardi 2019), and biochemistry (Zelesko et al. 2020). Very recently, pioneering work has started to develop generic frameworks for defining SGMs on arbitrary compact Riemannian manifolds (De Bortoli et al. 2022), and non-compact Riemannian manifolds (Huang et al. 2022).

In this work, instead of considering generic Riemannian manifolds, we are specifically concerned with the Special Orthogonal group in 3 dimensions, SO(3), which corresponds to the Lie group of 3D rotations. Modeling 3D orientations is of particularly high interest in many fields including for instance in robotics (estimating the pose of an object, Hoque et al. 2021); and in biochemistry (finding the conformation angle of molecules that minimizes the binding energy, Mansimov et al. 2019).

Contrary to more generic Riemannian manifolds, SO(3) benefits from specific properties, including a tractable heat kernel and efficient geometric ODE/SDE solvers, that will allow us to define very efficient diffusion models specifically for this manifold.

The contributions of our paper are summarized as follows:

- We reformulate Euclidean diffusion models on the SO(3) manifold, and demonstrate how the tractable heat kernel solution on SO(3) can be used to recover simple and efficient algorithms on this manifold.
- We provide concrete implementations of both Score-Based Generative Model and Denoising Diffusion Probabilistic Models specialized for SO(3).
- We reach a new state-of-the-art in sample quality on synthetic SO(3) distributions with our proposed SO(3) Score-Based Generative Model.
- We demonstrate the practicality and the expressive utility of our model in computer vision and astrophysics.

[1]For a comprehensive list of articles on score-based generative modeling, see https://scorebasedgenerativemodeling.github.io/

2 Diffusion Process on SO(3)

In this work, we are exclusively considering the SO(3) manifold, corresponding to the Lie group of 3D rotation matrices. We will denote by $\exp : \mathfrak{so}(3) \to$SO(3) and $\log :$ SO(3) $\to \mathfrak{so}(3)$ the exponential and logarithmic maps that connect SO(3) to its tangent space and Lie algebra $\mathfrak{so}(3)$.

Next, Similarly to Euclidean diffusion models (Song et al. 2021), we begin by defining a Brownian noising process that will be used to perturb the data distribution. Let us assume a Stochastic Differential Equation of the following form:

$$\mathrm{d}\mathbf{x} = \mathbf{f}(\mathbf{x}, t)\,\mathrm{d}t + g(t)\,\mathrm{d}\mathbf{w}, \tag{1}$$

where $\mathbf{w}$ is a Brownian process on SO(3), $\mathbf{f}(\cdot\,, t)$: SO(3) $\to$ $T_\mathbf{x}$SO(3) is a drift term, and $g(\cdot) : \mathbb{R} \to \mathbb{R}$ is a diffusion term, where $T_\mathbf{x}$SO(3) denotes the tangent space of SO(3). If we sample initial conditions for this SDE at $t = 0$ from a given data distribution $\mathbf{x}(0) \sim p_{\text{data}}$, we will denote by p_t the marginal distribution of $\mathbf{x}(t)$ at time $t > 0$. Thus $p_0 = p_{\text{data}}$, and at final time T at which we stop the diffusion process p_T will typically tend to a known target distribution that will be easy to sample from.

Just like in the Euclidean case, as demonstrated in De Bortoli et al. (2022), under mild regularity conditions 1 admits a reverse diffusion process on compact Riemannian manifolds such as SO(3), defined by the following reverse-time SDE:

$$\mathrm{d}\mathbf{x} = [\mathbf{f}(\mathbf{x}, t) - g(t)^2 \nabla \log p_t(\mathbf{x})]\mathrm{d}t + g(t)\mathrm{d}\bar{\mathbf{w}}, \tag{2}$$

where $\bar{\mathbf{w}}$ is a reversed-time Brownian motion and the *score function* $\nabla \log p_t(\mathbf{x}) \in T_\mathbf{x}$SO(3) is the derivative of the log marginal density of the forward process at time t. Corresponding to this reverse-time SDE, one can also define a probability flow ODE (Song et al. 2021):

$$\mathrm{d}\mathbf{x} = [\mathbf{f}(\mathbf{x}, t) - g(t)^2 \nabla \log p_t(\mathbf{x})]\mathrm{d}t. \tag{3}$$

This deterministic process is entirely defined once the score is known and maps p_T to any intermediate marginal distributions $\{p_t\}_{0 \leq t < T}$ of the forward process, including p_0. In particular, it can be seen as the equivalent of Neural ODE-based Continuous Normalizing Flows (CNF, Chen et al. 2018) with an explicit parameterization in terms of the score function. We illustrate this process in Figure 1 with samples from two Gaussian-like blobs on SO(3) being transported reversibly through this ODE between $t = 0$ and $t = T$.

While these equations are direct analog of the Euclidean SDEs and ODE described in (Song et al. 2021), defining diffusion generative models on SO(3) will mainly differ on the two following points:

- Defining the equivalent of the Gaussian heat kernel on SO(3): this is needed to easily sample from any intermediate p_t without having to simulate an SDE.
- Solving SDEs and ODEs on the manifold: contrary to the Euclidean case, the diffusion process must remained confined to the SO(3) manifold, which requires specific solvers.

We address these two points below before moving on to defining our generative models on SO(3).

2.1 The Isotropic Gaussian Distribution on SO(3)

In general, the main disadvantage of working on Riemannian manifolds compared to Euclidean space is that they lack a closed form expression for the heat kernel, i.e., the solution of the diffusion process (which is a Gaussian in Euclidean space). For compact manifolds, the heat kernel is in general only available as an infinite series, which in the case of SO(3), takes the following form (Nikolayev and Savyolov 1970):

$$f_\epsilon(\omega) = \sum_{\ell=0}^{\infty} (2\ell+1) \exp(-l(l+1)\epsilon^2) \frac{\sin((\ell + 1/2)\omega)}{\sin(\omega/2)} \tag{4}$$

where $\omega = |\boldsymbol{\omega}| \in (-\pi, \pi]$ is the rotation angle of the axis-angle representation $\boldsymbol{\omega}$ of a given rotation matrix and ϵ is a concentration parameter.

While for $\epsilon > 1$ this series converges quickly ($\ell_{\max} = 5$ is sufficient to achieve sub-percent accuracy), the convergence gets slower as ϵ gets smaller, which makes it impractical to model concentrated distributions. Thankfully, this series has been thoroughly studied in the literature and (Matthies, Muller, and Vinel 1988) shows that an excellent approximation of 4 can be achieved for $\epsilon < 1$ using the following closed-form expression:

$$f_\epsilon(\omega) \simeq \sqrt{\pi}\epsilon^{-3/2} e^{\frac{\epsilon}{4} - \frac{(\omega/2)^2}{\epsilon}} \left(\frac{\omega - e^{-\frac{\pi^2}{\epsilon}} \left((\omega - 2\pi) e^{\pi\omega/\epsilon} + (\omega + 2\pi) e^{-\pi\omega/\epsilon} \right)}{2 \sin(\omega/2)} \right) \tag{5}$$

Therefore, in practical applications, one can switch between using a truncation of eq. 4 for $\epsilon \geq 1$ and the approximation eq. 5 for $\epsilon < 1$.

Because of the property of being a solution of a diffusion process on SO(3), f_ϵ can be used to define the manifold equivalent of the Euclidean isotropic Gaussian distribution, which we will refer to as $\mathcal{IG}_{\text{SO(3)}}$, the Isotropic Gaussian on SO(3) (Leach et al. 2022; Ryu et al. 2022), also known in the literature as the normal distribution on SO(3) (Nikolayev and Savyolov 1970; Matthies, Muller, and Vinel 1988). For a given mean rotation $\boldsymbol{\mu} \in$ SO(3) and scale ϵ, the probability density of a rotation $\mathbf{x} \in$ SO(3) under $\mathcal{IG}_{\text{SO(3)}}(\boldsymbol{\mu}, \epsilon)$ is given by:

$$\mathcal{IG}_{\text{SO(3)}}(\mathbf{x}; \boldsymbol{\mu}, \epsilon) = f_\epsilon(\arccos\left[2^{-1}(\text{tr}(\boldsymbol{\mu}^T\mathbf{x}) - 1)\right])\,. \tag{6}$$

Sampling from $\mathcal{IG}_{\text{SO(3)}}(\boldsymbol{\mu}, \epsilon)$ is achieved in practice by inverse transform sampling. The cumulative distribution function over angles needed to sample with respect to the uniform distribution on SO(3) can be evaluated numerically given integrating $\frac{1-\cos(\omega)}{\pi} f_\epsilon(\omega)$ over $(-\pi, \pi]$. To form a rotation matrix $\mathbf{x} \sim \mathcal{IG}_{\text{SO(3)}}(\cdot; \boldsymbol{\mu}, \epsilon)$, one therefore first samples a rotation angle by inverse transform sampling given this CDF, then samples uniformly on $\mathcal{S}^2$ a rotation axis $\mathbf{v}$, yielding an axis-angle representation of a rotation matrix $\boldsymbol{\omega} = \omega\mathbf{v}$, which is then shifted by the mean of the distribution according to $\boldsymbol{x} = \boldsymbol{\mu} \exp(\boldsymbol{\omega})$.

An important property of $\mathcal{IG}_{\text{SO(3)}}(\mu, \epsilon)$, which sets it apart from other distributions on SO(3) (e.g. Bingham, Matrix

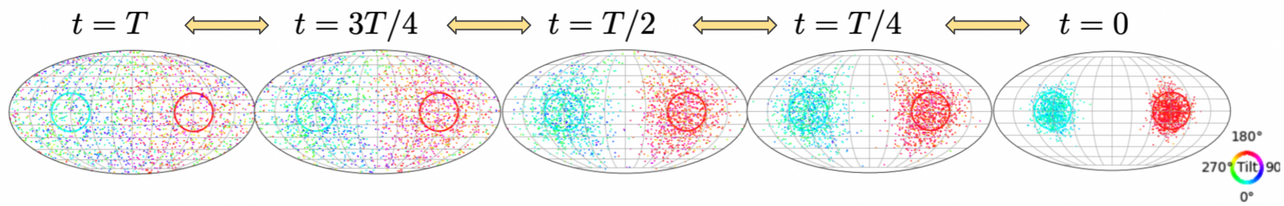

Figure 1: Illustration of reversible diffusion of a mixture of two $\mathcal{IG}_{\mathrm{SO(3)}}$ blobs on SO(3). Samples from a given base distribution (right most, denoted by circles) can be evolved under the probability flow ODE (Eq. 3) towards a noisy distribution (left most), or vice-versa from the noisy distribution back to the target distribution. Each point represents a rotation matrix in SO(3) projected on the sphere according to its canonical axis, the color indicates the tilt around that axis (visualisation adopted from Murphy et al. 2021). An animation of this figure is available*

Algorithm 1: Geometric ODE solver on SO(3) (Heun's method) for $\mathrm{d}\mathbf{x} = \mathbf{f}(\mathbf{x}, t)\,\mathrm{d}t$

Require: Step size h, initial condition $\mathbf{x}_0$, time steps $\{t_n\}_{n=0}^N$, number of steps N

1: **for** $n \in \{0, \dots, N-1\}$ **do**
2: $\quad \mathbf{y}_1 = h\,\mathbf{f}(\mathbf{x}_n,\, t_n)$
3: $\quad \mathbf{y}_2 = h\,\mathbf{f}(\exp(\frac{1}{2}\mathbf{y}_1)\mathbf{x}_n,\, t_n + \frac{1}{2}h)$
4: $\quad \mathbf{x}_{n+1} = \exp(\mathbf{y}_2)\mathbf{x}_n$
5: **end for**
6: **return** $\{\mathbf{x}_n\}_{n=0}^N$

Fisher, Wrapped Normal, more on this in Appendix E.1), is that it remains *closed under convolution*, as a direct consequence of being the solution of a diffusion process. The convolution of two centered $\mathcal{IG}_{\mathrm{SO(3)}}$ distributions of scale parameter ϵ_1 and ϵ_2 is an $\mathcal{IG}_{\mathrm{SO(3)}}$ distribution of scale $\epsilon_1 + \epsilon_2$.

We will also note two interesting asymptotic behaviors. For large ϵ, it tends to $\mathcal{U}_{\mathrm{SO(3)}}$, the uniform distribution on SO(3), while for small ϵ the distribution $\mathcal{IG}_{\mathrm{SO(3)}}(\mathbf{I}, \epsilon)$ can locally be approximated in the axis-angle representation of the tangent space by a normal distribution $\mathcal{N}(0, \sigma^2\mathbf{I})$ in $\mathbb{R}^3$, with $\epsilon = \frac{\sigma^2}{2}$.

2.2 Solving ODEs on SO(3)

Thanks to the existence of a tractable heat kernel on SO(3), the generative models we will define in the next section will not actually require us to solve the SDEs introduced at the beginning of this section, and we will only need to solve the probability flow ODE defined in eq. 3.

Solving differential equations on manifolds can broadly be achieved using two distinct strategies, either projection methods using a Euclidean solver followed by a projection step onto the manifold, or intrinsic methods that rely on additional structure of the manifold to define an iteration that remains by construction on the manifold. In this work, we are concerned with SO(3), which is not only a compact Riemannian manifold, but also possesses a Lie group structure, which makes it amenable to efficient solvers. In particular, we will make use of the Runge-Kutta-Munthe-Kaas (RK-MK) class of algorithms and direct the interested reader to (Iserles et al. 2000) for a review of Lie group integrators. We adopt in practice the Lie group equivalent of Heun's method, which is one variant of RK-MK integrators, and we provide the details of this integrator in Algorithm 1.

While we will not require it in practice, it is also possible to build SDE solvers on SO(3) with a similar strategy, and we point the interested reader for instance to the Geodesic Random Walk algorithm described in (De Bortoli et al. 2022).

3 Diffusion Generative Models on SO(3)

In this section, we present two different approaches to building generative models based on the forward and reverse diffusion processes presented in Section 2 resulting in SO(3) specific Score-Based Generative Models, and Denoising Diffusion Probabilistic Models.

3.1 Score-based Generative Model

Although various choices for the particular form of the forward SDE in eq. 1 are possible, for simplicity (and without loss of generality) we adopt in this section the so-called Variance-Exploding SDE, which is the canonical choice of the Euclidean Score-Matching Langevin Dynamics (Song and Ermon 2019). More specifically, we define $\mathbf{f}(\mathbf{x}, \mathbf{t}) = 0$ and $g(t) = \sqrt{\frac{\mathrm{d}\epsilon(t)}{\mathrm{d}t}}$ for a given choice of noise schedule $\epsilon(t)$, which reduces eq. 1 to:

$$\mathrm{d}\mathbf{x} = \sqrt{\frac{\mathrm{d}\epsilon(t)}{\mathrm{d}t}}\,\mathrm{d}\mathbf{w}\ . \tag{7}$$

For our fiducial model, and unless stated otherwise, we will further assume for simplicity the following noise schedule: $\epsilon(t) = t$. The main drawback of this SDE in Euclidean geometry is that it will tend to a Gaussian with infinitely large variance. However, on SO(3) this SDE will tend to the uniform distribution $\mathcal{U}_{\mathrm{SO(3)}}$, which is a natural choice for the prior distribution at large T.

Following from this choice of SDE, we can define a noise kernel $p_\epsilon(\tilde{\boldsymbol{x}}|\boldsymbol{x}) = \mathcal{IG}_{\mathrm{SO(3)}}(\tilde{\boldsymbol{x}}; \boldsymbol{x}, \epsilon)$ for $\boldsymbol{x}, \tilde{\boldsymbol{x}} \in$SO(3), such that the data distribution convolved by this noise kernel becomes

$$p_\epsilon(\boldsymbol{x}) = \int_{SO(3)} p_{\text{data}}(\boldsymbol{x}')p_\epsilon(\boldsymbol{x}|\tilde{\boldsymbol{x}})\,\mathrm{d}\boldsymbol{x}\ , \tag{8}$$

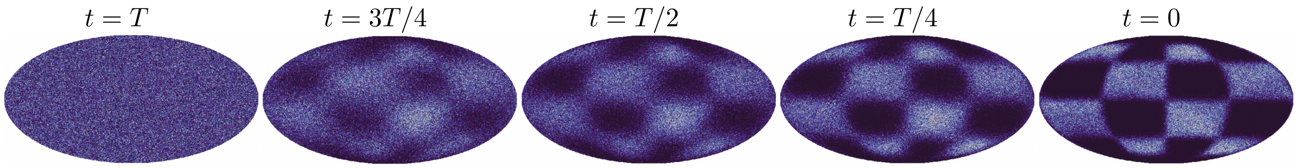

Figure 2: Sampling from a Diffusion Generative Model trained on a synthetic density on SO(3). Starting from $\mathcal{U}_{SO(3)}$, the uniform distribution on SO(3) at $t = T$ (left), the sampling procedure (either based on SGMs or DDPMs) denoises this distribution back to the target density at $t = 0$ (right). For visualization this density plot shows the distribution of canonical axes of sampled rotations projected on the sphere; the tilt around that axis is discarded.

and corresponds to p_t, the marginal distribution of the diffusion process at time t: $p_{\epsilon(t)} = p_t$.

This strategy directly extends Euclidean SGMs (Song and Ermon 2019; Song et al. 2021) and relies on the time-reversed diffusion process described in eq. 2. Samples from the learned distribution p_0 can be sampled by first sampling $\mathbf{x}_T \sim \mathcal{U}_{SO(3)}$ and evolving these samples either through the reverse SDE (eq. 2) or probability flow ODE (eq. 3) back to $t = 0$. This process is entirely defined as soon as the *score function* of the marginal distribution at any intermediate time t, $\nabla \log p_{\epsilon(t)}$, is known. Therefore the first step is to establish a score-matching strategy on SO(3).

Let us consider $\{X_i\}_{i=0}^3$, an orthonormal basis of the tangent space $T_{\mathbf{e}}$SO(3). The directional derivative of the log density of the noise kernel $p_\epsilon(\boldsymbol{x}|\tilde{\boldsymbol{x}})$ can be computed as:

$$\nabla_{X_i} \log p_\epsilon(\tilde{\mathbf{x}}|\mathbf{x}) = \frac{\mathrm{d}}{\mathrm{d}s} \log p_\epsilon(\tilde{\mathbf{x}} \exp(sX_i)|\mathbf{x}) \bigg|_{s=0} , \quad (9)$$

which can be computed in practice by automatic differentiation given the explicit approximation formulae for the $\mathcal{IG}_{\text{SO(3)}}$ distribution introduced in Section 2.1. To match this derivative, we introduce a neural score estimator $s_\theta(\mathbf{x}, \epsilon)$: SO(3) $\times \mathbb{R}^{+\star} \to \mathbb{R}^3$, which can be trained directly under a conventional denoising score matching loss:

$$\mathcal{L}_{DSM} = \mathbb{E}_{p_{\text{data}}(\mathbf{x})} \mathbb{E}_{\epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2)} \mathbb{E}_{p_{|\epsilon|}(\tilde{\mathbf{x}}|\mathbf{x})} \left[|\epsilon| \ \| \ s_\theta(\tilde{\mathbf{x}}, \epsilon) - \nabla_X \log p_{|\epsilon|}(\tilde{\mathbf{x}}|\mathbf{x}) \ \|_2^2 \right]$$

where we sample at training time random noise scales $\epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2)$ similarly to (Song and Ermon 2020). The minimum of this loss will be achieved for $s_\theta(\mathbf{x}, \epsilon) = \nabla \log p_\epsilon$.

Once the score function is estimated from data using this score matching loss, sampling from the generative model can be achieved by using the reverse SDE formula, or using the ODE flow formula. In this work, we use the latter for its simplicity and speed, so that our specific fiducial sampling strategy becomes:

$$\mathbf{x}_T \sim \mathcal{U}_{\text{SO(3)}} \quad ; \quad \mathrm{d}\mathbf{x}_t = -\frac{1}{2} \frac{\mathrm{d}\epsilon(t)}{\mathrm{d}t} s_\theta(\mathbf{x}_t, \epsilon(t)) \, \mathrm{d}t \quad (10)$$

which we solve down to $t = 0$ with the geometric ODE solver described in Algorithm 1. Compared to stochastic sampling strategies based on simulating the reverse SDE, this approach has several advantages. 1) It is much faster, and can benefit from adaptive ODE solvers bringing down the number of score evaluations needed, 2) the same ODE can be used to evaluate the log likelihood of the model by applying the probability flow formula of CNFs.

Algorithm 2: Sampling from Denoising Diffusion Probabilistic Model on SO(3)

Require: Trained neural networks $\mu_\theta(\mathbf{x}, t), \epsilon_\theta(\mathbf{x}, t)$, number of steps N, time steps $\{t_i\}_{i=0}^N$

```
x_N ~ IG_SO(3)(I, 1)
for i = {N, N-1, ..., 1} do
    x_{i-1} ~ p_θ(·; x_i) = IG_SO(3)(·; μ_θ(x_i, t_i), ε_θ(x_i, t_i))
end for
return {x_n}_{n=0}^N
```

3.2 Denoising Diffusion Probabilistic Model

Similarly to the previous section, although several forms for the forward SDE in eq. 1 are possible, to define a DDPM we adopt the canonical choice of Ho, Jain, and Abbeel (2020); Sohl-Dickstein et al. (2015) of a Variance-Preserving SDE defined as:

$$\mathrm{d}\mathbf{x} = -\frac{1}{2}\beta(t)\mathbf{x} \, \mathrm{d}t + \sqrt{\beta(t)} \, \mathrm{d}\mathbf{w}, \quad (11)$$

with $\beta(t)$ a function of time with values in $(0, 1)$. This SDE will tend to a standard $\mathcal{IG}_{\text{SO(3)}}$

As described in Song et al. (2021), when using a finite number of steps, the forward diffusion process defined by eq. 11 $\{\mathbf{x}_i\}_{i=0}^N$ (corresponding to times $\{0 \leq t_i \leq T\}_{i=0}^n$) can be interpreted as a Markov process:

$$p(\mathbf{x}_{0:N}) = p(\mathbf{x}_0) p_{\beta_1}(\mathbf{x}_1|\mathbf{x}_0) \dots p_{\beta_N}(\mathbf{x}_N|\mathbf{x}_{N-1}) \quad (12)$$

with the transition kernel $p_{\beta_{i+1}}(\mathbf{x}_{i+1}|\mathbf{x}_i) = \mathcal{IG}_{\text{SO(3)}}(\mathbf{x}_{i+1}; \sqrt{1-\beta_{i+1}}\mathbf{x}_i, \beta_{i+1})$. The idea of DDPMs is to introduce a reverse Markov process defined in terms of variational transition kernels $p_\theta(\boldsymbol{x}_{i-1}|\boldsymbol{x}_i)$:

$$p_\theta(\mathbf{x}_{0:N}) = p_\theta(\boldsymbol{x}_N) p_\theta(\mathbf{x}_{N-1}|\mathbf{x}_N) \dots p_\theta(\mathbf{x}_0|\mathbf{x}_1). \quad (13)$$

While one could choose any distribution on SO(3) to parameterize this inverse transition kernel (e.g., Matrix Fisher, Bingham), we adopt for convenience an Isotropic Gaussian on SO(3) and use the following expression:

$$p_\theta(\boldsymbol{x}_{i-1}|\boldsymbol{x}_i) = \mathcal{IG}_{\text{SO(3)}}(\boldsymbol{x}_{i-1}; \boldsymbol{x}_i \, \boldsymbol{\delta}_\theta(\boldsymbol{x}_i, t_i), \epsilon_\theta(\boldsymbol{x}_i, t_i)) \quad (14)$$

where $\boldsymbol{\delta}_\theta : \mathrm{SO}(3) \times \mathbb{R}^+ \rightarrow \mathrm{SO}(3)$ is a neural network predicting the residual rotation to apply to $\boldsymbol{x}_i$ to obtain the mean of the reverse kernel and $\epsilon_\theta : \mathrm{SO}(3) \times \mathbb{R}^+ \rightarrow \mathbb{R}^+$ is a neural network predicting the variance of this reverse kernel. To parameterize the output of $\boldsymbol{\delta}_\theta$ we adopt the 6D continuous rotation representation of (Zhou et al. 2019) and explore the impact of this choice in Section D.

If the reverse Markov process can be successfully trained to match the forward process, it provides a direct sampling strategy to generate samples from p_0 by initializing the chain from p_T and iteratively sampling from the reverse kernel $p_\theta(\boldsymbol{x}_{i-1}|\boldsymbol{x}_i)$. In DDPMs, the training strategy is to write down the Evidence Lower Bound (ELBO), given this variational approximation for the reverse Markov process, in order to train the individual transition kernels $p_\theta(\mathbf{x}_{i-1}|\mathbf{x}_i)$. To reduce the variance of this loss over a naive evaluation of the ELBO, (Sohl-Dickstein et al. 2015) and (Ho, Jain, and Abbeel 2020) propose to use a closed form expression of the reverse kernel $p(\mathbf{x}_{i-1}|\mathbf{x}_i, \mathbf{x}_0)$ when conditioned on $\mathbf{x}_0$. This makes it possible to rewrite the ELBO in terms of analytic KL divergences between Gaussian transitions kernels. However, contrary to the Gaussian case of Euclidean DDPMs, for $\mathcal{IG}_{\mathrm{SO}(3)}$ we do not easily have access to a closed form expression of the reverse kernel $p(x_{t-1}|x_t, x_0)$ which is needed to derive the training loss used in (Ho, Jain, and Abbeel 2020). The same approach cannot be applied.

Instead, we consider the expression for the ELBO:

$$\mathbb{E}\left[-\log p_\theta(\mathbf{x}_0)\right] \leq$$
$$\mathbb{E}_p\left[-\log p(\mathbf{x}_N) - \sum_{i\geq 1} \log \frac{p_\theta(\mathbf{x}_{i-1}|\mathbf{x}_i)}{p(\mathbf{x}_i|\mathbf{x}_{i-1})}\right] =: \mathcal{L}_{\mathrm{ELBO}}$$

which will be optimized by maximizing the log likelihood of individual transition kernels $\log p_\theta(\mathbf{x}_{i-1}|\mathbf{x}_i)$ over samples $\mathbf{x}_{i-1}, \mathbf{x}_i$ obtained through simulating the forward Markov diffusion process over the training set. Our strategy on SO(3), is therefore to train each transition kernel by maximum likelihood using the following loss function:

$$\mathcal{L}_{DDPM} :=$$
$$\sum_{i\geq 0} \mathbb{E}_{p_{\text{data}}(\mathbf{x}_0)} \mathbb{E}_{p_\epsilon(\mathbf{x}_i|\mathbf{x}_0)} \mathbb{E}_{p_{\epsilon_i}(\mathbf{x}_{i+1}|\mathbf{x}_i)} \left[-\log p_\theta(\mathbf{x}_i|\mathbf{x}_{i+1})\right]$$

where the log probability of the $\mathcal{IG}_{\mathrm{SO}(3)}$ distribution used in our parameterised reverse kernel is defined in eq. 6. While this loss can indeed be used to train a DDPM (as demonstrated in the next section), compared to the strategy of (Ho, Jain, and Abbeel 2020), we expect it to suffer from larger variance and is not explicitly parameterised in terms of the score function (Song et al. 2021). Once trained, we can use the sampling strategy described in Algorithm 2 to draw from the generative model.

4 Experiments

4.1 Test Densities on SO(3)

We adopt three different toy distributions on SO(3): a checkerboard pattern, a multi-modal distribution of 4 concentrated Gaussians and a stripe pattern that can be viewed as circles on the sphere. We focus on evaluating the generative models in terms of the quality of their sample generation using the Classifier 2-Sample Tests (C2ST) metric (Lopez-Paz and Oquab 2017; Dalmasso et al. 2020; Lueckmann et al. 2021). The C2ST metric has been used in particular in the context of simulation-based inference to quantify the quality of inferred distributions. Concisely, the C2ST method uses a neural network classifier to discriminate between true and the generated samples, yielding a value of 0.5 if the two distributions are perfectly indistinguishable to the classifier, up to a value of 1 if they are extremely different. In contrast to the usual Negative Log Likelihood (NLL), C2ST can be consistently computed for all generative models we compare bellow.

We present in Figure 3 and Table 1 the results of our comparisons on these test densities against the implicit-pdf method of Murphy et al. (2021), the DDPM implementation of Leach et al. (2022), Moser flow of Rozen et al. (2021), and the Riemannian Score-Based Generative Model (RSGM) of De Bortoli et al. (2022) (trained under their $\ell_{t|0}$ score matching loss). We find that in all cases our SGM implementation on SO(3) yields the best C2ST metric, which is in line with the visual quality of distributions shown in Figure 3. Our DDPM implementation on SO(3) yields distributions that are comparatively less sharp, which we attribute to the larger variance of our training loss for that model. Compared to other models, our experiments illustrate a failure mode in the method of Leach et al. (2022) which we attribute to the fact that the usual DDPM loss function cannot be directly translated to SO(3) (as discussed in Section 3.2). We also note that the Implicit-PDF model, in comparison, is extremely limited in resolution because of the memory cost of evaluating the pdf on a tiling of SO(3), and thus yields much lower scores. The best results after our method are achieved by the RSGM model (De Bortoli et al. 2022), which is expected due to its similarity with our work, but is slower to train in the specific case of SO(3). We find that the cost of simulating the forward SDE in the training phase leads to a factor x8 in computation time per batch on a given GPU.

4.2 Pose Estimation

To test practical applications of our model, following Murphy et al. (2021) we used a vision description obtained from a pre-trained ResNet architecture with ImageNet weights consisting of 2048 dimensional vector to condition an SO(3) SGM. Conditioning the model on visual descriptors can be easily done by concatenating it to the input of the score network. Using images of symmetric solids from the SYMSOL dataset (Murphy et al. 2021) we show that we can correctly estimate poses of objects with degenerate symmetry, as shown in Figure 4. The model captures the underlying density well, as illustrated by column 4 where the probabilities are almost uniform along the density path. Additionally, uncertainty regions are expressed by the model as shown in column 3. Here we specifically chose symmetric objects in order to test the practical expressivity of our model. Pose estimation is more challenging for symmetric objects than for objects that lack

Model	Checkerboard	4-Gaussians	3-Stripes
SGM on SO(3) (ours)	**0.50**$\pm$ 0.01	**0.50**$\pm$ 0.01	**0.51**$\pm$ 0.01
DDPM on SO(3) (ours)	0.52$\pm$ 0.01	0.53$\pm$ 0.01	0.52$\pm$ 0.01
RSGM (De Bortoli et al. 2022)	0.51$\pm$ 0.01	–	**0.51** $\pm$ 0.01
Moser Flow (Rozen et al. 2021)	0.56$\pm$ 0.01	0.60$\pm$ 0.02	0.53$\pm$ 0.02
DDPM (Leach et al. 2022)	0.71$\pm$ 0.04	0.90$\pm$ 0.05	0.60$\pm$ 0.03
Implicit-PDF (Murphy et al. 2021)	0.59$\pm$ 0.04	0.81$\pm$ 0.09	0.63$\pm$ 0.04

Table 1: Sample quality metric from the C2ST (lower is better). If the learned distribution is identical to the original one, the metric should be ~ 0.5; if it is significantly different, the metric tends towards ~ 1. The errors on the metric were obtain from the standard deviation of the metric over k-fold cross validation samples for a single training of the model. – indicate a failure to evaluate the metric for a particular model.

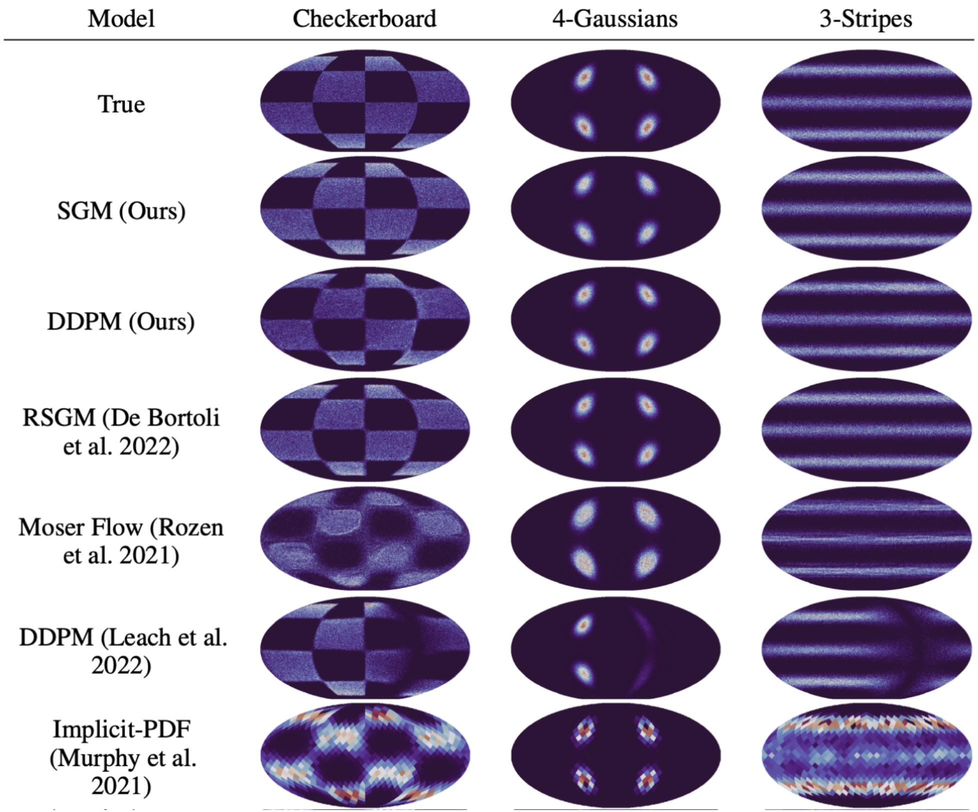

Figure 3: Density plot comparing samples from learned synthetic densities on SO(3). For visualization this density plot shows the distribution of canonical axes of sampled rotations projected on the sphere; the tilt around that axis is discarded.

symmetry, for which the process boils down to learning and predicting a single point on SO(3) manifold. Compared to the Implicit-PDF results presented in Murphy et al. (2021), our approach necessitates far less memory, all the while reaching higher quality density estimates.

4.3 Conditional Galaxy Orientation Modeling

In this experiment, we are interested in emulating the behavior of galaxies in numerical simulations of the Universe that follow the formation and evolution of galaxies over cosmological times (from few hundred million years after the Big Bang down to modern day). These simulations are computationally costly and even intractable at scales and resolutions demanded by future cosmological surveys. Therefore, emulation of galaxy properties using Machine Learning methods could potentially significantly speed up analysis pipelines in cosmology. The particular property we want to model is the 3D orientation of galaxies with respect to their surrounding environment. This important effect, called Intrinsic Alignment, relates to the fact that galaxies are not randomly oriented in the Universe, but tend to gain some preferential alignments through a variety of physical processes during their formation and evolution process.

The problem can be formulated as a conditional density

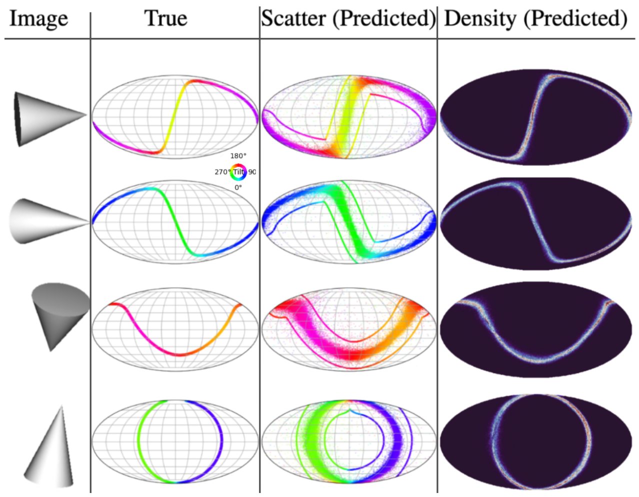

Figure 4: Predicted poses for an image of a solid with degenerate symmetry; here we only show results for a cone from multiple angles. The 1st column depicts the image of the symmetric solid and the 2nd column shows the underlying true pose of the object. In column 3, each point represents a rotation matrix in SO(3) projected on the sphere according to its canonical axis, with the color indicating the tilt around that axis. For visualization purposes, the density plot (column 4) shows the distribution of canonical axes of sampled rotations projected on the sphere; the tilt around that axis is discarded.

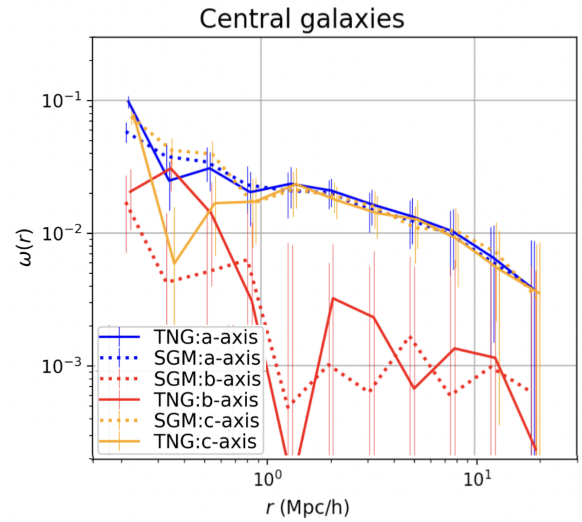

Figure 5: The correlation function, $\omega(r)$, which captures the correlation between position and the axis direction, of all galaxy axes with galaxy positions: the solid lines show the measured values from the TNG simulation, while the dashed lines show the generated values from the SGM. The SGM curve was shifted by 5 per cent to the left for visual clarity. For the ellipsoid, we denote the major, intermediate, and minor axes as a, b, and c, respectively. Error bars were obtained using the jackknife method.

estimation problem over SO(3) given some summary information about their local environment, such as the local gravitational tidal field: given by the 3D tidal tensor $\mathbf{T}$ (more in Appendix G):

$$p(\mathbf{x}_{\text{galaxy}}|\mathbf{T}) \tag{15}$$

where $\mathbf{x}_{\text{galaxy}} \in$ SO(3) are the correlated galaxy orientations conditioned on the tidal field $\mathbf{T}$ around the galaxies. We want to model this density with a conditional score generative network $s_\theta(\mathbf{x}_{\text{galaxy}}|\mathbf{T})$ as such:

$$\nabla_{\mathbf{x}_{\text{galaxy}}} \log \left[p(\mathbf{x}_{\text{galaxy}}|\mathbf{T})\right] \approx \mathbf{s}_\theta(\mathbf{x}_{\text{galaxy}}|\mathbf{T}) \tag{16}$$

Results For this experiment we use a state-of-the-art simulation, IllustrisTNG, (more in Appendix F), where we model galaxies as 3D ellipsoids (as it is conventional in astrophysics/cosmology), retrieve their 3D orientation and compute the summary information about their environments using $\mathbf{T}$. Throughout the section we refer to the sample generated from the diffusion model as the *SGM* sample, and the sample from IllustrisTNG as the *TNG* sample. The inputs to the model are the gravitational tidal field (obtained from the 3D tidal tensor which carries some information about the alignment at large scales), and the outputs are the 3D orientations of galaxies: the model generates the orientations of galaxies conditioned on the tidal field.

In order to quantify how robustly we can recover the correct conditional density of orientation, we use a domain-specific quantitative measurement of correlation between galaxy orientations as a function of their respective distance – $\omega(r)$ (more in Appendix H), and show very good agreement between the orientation of galaxies with respect to the surrounding environment. This correlation function captures the correlation between the large scale distribution of matter in the simulated volume (galaxy positions) and orientations of the selected galaxy axes (modeling the galaxies as ellipsoids and selecting either the major, intermediate, or minor axis). Positive $\omega(r)$ values indicate that the selected galaxy axis exhibits a coherent alignment towards the positions of nearby galaxies. The $\omega(r)$ correlation functions for all three axes of the galaxies are presented in Fig. 5. In general, the qualitative trend of $\omega(r)$ as a function of 3D separation is captured by the SGM. For small separation, there is a general deviation from the measured values, which may be explained by the highly complex hydrodynamical processes that might not have been captured by the neural network; or the tidal field does not contain enough information. Quantitatively, for the major and minor axes the generated samples agree well with the simulation. For the intermediate axes, the signal is very weak, though the SGM managed to captured the correlation with statistical consistency. Overall, the SGM model can describe synthetic densities with high statistical correlations , and those with low statistical correlations, as shown in the case of galaxy alignments.

5 Conclusions and Limitations

In this paper, we have presented a framework for score-based diffusion generative models on SO(3), as an extension of Euclidean SDE-based models (Song et al. 2021). Because it is developed specifically for the SO(3) manifold, our work proposes a simpler and more efficient alternative to other recent (and general) Riemannian diffusion models while reaching state-of-the-art quality on synthetic distributions on SO(3). Additionally, we presented pose estimation results by conditioning the model on visual descriptors using the SYMSOL dataset. Our model correctly predicts the poses of objects with degenerate symmetry with low memory cost. One of the most promising applications of this work is in robotics and computer vision, for the general task of pose-estimation, where our proposed model significantly outperforms current baselines (Murphy et al. 2021). As for applications in the natural sciences, the generative model robustly captures low-signal statistical correlations of galaxy alignments in a state-of-the-art cosmological simulation.

Acknowledgements

This work was supported in part by a grant from the Simons Foundation (Simons Investigator in Astrophysics, Award ID 620789) and by the NSF AI Institute: Physics of the Future, NSF PHY- 2020295.

References

Ben-Hamu, H.; Cohen, S.; Bose, J.; Amos, B.; Nickel, M.; Grover, A.; Chen, R. T. Q.; and Lipman, Y. 2022. Matching Normalizing Flows and Probability Paths on Manifolds. In Chaudhuri, K.; Jegelka, S.; Song, L.; Szepesvari, C.; Niu, G.; and Sabato, S., eds., *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, 1749–1763. PMLR.

Brehmer, J.; and Cranmer, K. 2020. Flows for Simultaneous Manifold Learning and Density Estimation. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20. Red Hook, NY, USA: Curran Associates Inc. ISBN 9781713829546.

Cao, N. D.; and Aziz, W. 2020. The Power Spherical distribution. *ArXiv*, abs/2006.04437.

Chen, R. T. Q.; Rubanova, Y.; Bettencourt, J.; and Duvenaud, D. K. 2018. Neural Ordinary Differential Equations. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Craven, S.; Croon, D.; Cutting, D.; and Houtz, R. 2022. Machine learning a manifold. *Phys. Rev. D*, 105: 096030.

Dalmasso, N.; Lee, A.; Izbicki, R.; Pospisil, T.; Kim, I.; and Lin, C.-A. 2020. Validation of Approximate Likelihood and Emulator Models for Computationally Intensive Simulations. In Chiappa, S.; and Calandra, R., eds., *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, 3349–3361. PMLR.

De Bortoli, V.; Mathieu, E.; Hutchinson, M.; Thornton, J.; Teh, Y. W.; and Doucet, A. 2022. Riemannian Score-Based Generative Modeling. ArXiv:2202.02763 [cs, math, stat].

Falorsi, L.; de Haan, P.; Davidson, T. R.; and Forré, P. 2019. Reparameterizing Distributions on Lie Groups. ArXiv:1903.02958 [cs, math, stat].

Falorsi, L.; and Forré, P. 2020. Neural Ordinary Differential Equations on Manifolds. ArXiv:2006.06663 [cs, stat].

Feiten, W.; Lang, M.; and Hirche, S. 2013. Rigid motion estimation using mixtures of projected Gaussians. In *Proceedings of the 16th International Conference on Information Fusion*, 1465–1472.

Gaddes, M. E.; Hooper, A.; and Bagnardi, M. 2019. Using Machine Learning to Automatically Detect Volcanic Unrest in a Time Series of Interferograms. *Journal of Geophysical Research: Solid Earth*, 124(11): 12304–12322.

Gemici, M. C.; Rezende, D.; and Mohamed, S. 2016. Normalizing Flows on Riemannian Manifolds. ArXiv:1611.02304 [cs, math, stat].

Gilitschenski, I.; Sahoo, R.; Schwarting, W.; Amini, A.; Karaman, S.; and Rus, D. 2020. Deep Orientation Uncertainty Learning based on a Bingham Loss. In *International Conference on Learning Representations*.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative Adversarial Nets. In Ghahramani, Z.; Welling, M.; Cortes, C.; Lawrence, N.; and Weinberger, K., eds., *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.

Grigoryan, A. 2009. *Heat Kernel and Analysis on Manifolds*. AMS/IP studies in advanced mathematics. American Mathematical Society. ISBN 9780821849354.

Hartley, R. I.; Trumpf, J.; Dai, Y.; and Li, H. 2013. Rotation Averaging. *International Journal of Computer Vision*, 103(3): 267–305.

Hemmati, S.; Capak, P.; Pourrahmani, M.; Nayyeri, H.; Stern, D.; Mobasher, B.; Darvish, B.; Davidzon, I.; Ilbert, O.; Masters, D.; and Shahidi, A. 2019. Bringing Manifold Learning and Dimensionality Reduction to SED Fitters. *The Astrophysical Journal*, 881(1): L14.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising Diffusion Probabilistic Models. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 6840–6851. Curran Associates, Inc.

Hoque, S.; Arafat, M. Y.; Xu, S.; Maiti, A.; and Wei, Y. 2021. A Comprehensive Review on 3D Object Detection and 6D Pose Estimation With Deep Learning. *IEEE Access*, 9: 143746–143770.

Huang, C.-W.; Aghajohari, M.; Bose, A. J.; Panangaden, P.; and Courville, A. 2022. Riemannian Diffusion Models. ArXiv:2208.07949 [cs].

Iserles, A.; Munthe-Kaas, H. Z.; Nørsett, S. P.; and Zanna, A. 2000. Lie-group methods. *Acta Numerica*, 9: 215–365.

Jagvaral, Y.; Lanusse, F.; Singh, S.; Mandelbaum, R.; Ravanbakhsh, S.; and Campbell, D. 2022. Galaxies and haloes on graph neural networks: Deep generative modelling scalar and vector quantities for intrinsic alignment. *Monthly Notices of the Royal Astronomical Society*, 516(2): 2406–2419.

Kingma, D. P.; and Welling, M. 2013. Auto-Encoding Variational Bayes. *arXiv e-prints*, arXiv:1312.6114.

Leach, A.; Schmon, S. M.; Degiacomi, M. T.; and Willcocks, C. G. 2022. Denoising Diffusion Probabilistic Models on SO (3) for Rotational Alignment. In *ICLR 2022 Workshop on Geometrical and Topological Representation Learning*.

Lee, J.; Springel, V.; Pen, U.-L.; and Lemson, G. 2008. Quantifying the cosmic web - I. The large-scale halo ellipticity-ellipticity and ellipticity-direction correlations. *Monthly Notices of the Royal Astronomical Society*, 389(3): 1266–1274.

Lopez-Paz, D.; and Oquab, M. 2017. Revisiting Classifier Two-Sample Tests. In *International Conference on Learning Representations*.

Lueckmann, J.-M.; Boelts, J.; Greenberg, D. S.; Gonçalves, P. J.; and Macke, J. H. 2021. Benchmarking Simulation-Based Inference. In *AISTATS*, 343–351.

Mansimov, E.; Mahmood, O.; Kang, S.; and Cho, K. 2019. Molecular Geometry Prediction using a Deep Generative Graph Neural Network. *Scientific Reports*, 9.

Marinacci, F.; et al. 2018. First results from the IllustrisTNG simulations: radio haloes and magnetic fields. *Mon. Not. Roy. Astron. Soc.*, 480(4): 5113–5139.

Mathieu, E.; and Nickel, M. 2020. Riemannian Continuous Normalizing Flows. ArXiv:2006.10605 [cs, stat].

Matthies, S.; Muller, J.; and Vinel, G. W. 1988. On the Normal Distribution in the Orientation Space. *Textures and Microstructures*, 10: 77–96.

Mohlin, D.; Bianchi, G.; and Sullivan, J. 2020. Probabilistic orientation estimation with matrix Fisher distributions. ArXiv:2006.09740 [cs].

Murphy, K. A.; Esteves, C.; Jampani, V.; Ramalingam, S.; and Makadia, A. 2021. Implicit-PDF: Non-Parametric Representation of Probability Distributions on the Rotation Manifold. In *International Conference on Machine Learning*, 7882–7893. PMLR.

Naiman, J. P.; Pillepich, A.; Springel, V.; Ramirez-Ruiz, E.; Torrey, P.; Vogelsberger, M.; Pakmor, R.; Nelson, D.; Marinacci, F.; Hernquist, L.; Weinberger, R.; and Genel, S. 2018. First results from the IllustrisTNG simulations: a tale of two elements - chemical evolution of magnesium and europium. *Monthly Notices of the Royal Astronomical Society*, 477(1): 1206–1224.

Nelson, D.; Springel, V.; Pillepich, A.; Rodriguez-Gomez, V.; Torrey, P.; Genel, S.; Vogelsberger, M.; Pakmor, R.; Marinacci, F.; Weinberger, R.; Kelley, L.; Lovell, M.; Diemer, B.; and Hernquist, L. 2019. The IllustrisTNG simulations: public data release. *Computational Astrophysics and Cosmology*, 6(1): 2.

Nelson, D.; et al. 2018. First results from the IllustrisTNG simulations: the galaxy colour bimodality. *Mon. Not. Roy. Astron. Soc.*, 475(1): 624–647.

Nikolayev, D. I.; and Savyolov, T. I. 1970. Normal distribution on the rotation group SO (3). *Textures and Microstructures*, 29.

Peretroukhin, V.; Giamou, M.; Greene, W. N.; Rosen, D.; Kelly, J.; and Roy, N. 2020. A Smooth Representation of Belief over SO(3) for Deep Rotation Learning with Uncertainty. In *Proceedings of Robotics: Science and Systems*. Corvalis, Oregon, USA.

Pillepich, A.; Nelson, D.; Hernquist, L.; Springel, V.; Pakmor, R.; Torrey, P.; Weinberger, R.; Genel, S.; Naiman, J. P.; Marinacci, F.; and Vogelsberger, M. 2018. First results from the IllustrisTNG simulations: the stellar mass content of groups and clusters of galaxies. *Monthly Notices of the Royal Astronomical Society*, 475(1): 648–675.

Rezende, D.; and Mohamed, S. 2015. Variational Inference with Normalizing Flows. In Bach, F.; and Blei, D., eds., *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, 1530–1538. Lille, France: PMLR.

Rozen, N.; Grover, A.; Nickel, M.; and Lipman, Y. 2021. Moser Flow: Divergence-based Generative Modeling on Manifolds. ArXiv:2108.08052 [cs, stat].

Ryu, H.; Lee, J.-H.; Lee, H.-i.; and Choi, J. 2022. Equivariant Descriptor Fields: SE(3)-Equivariant Energy-Based Models for End-to-End Visual Robotic Manipulation Learning. ArXiv:2206.08321 [cs].

Sohl-Dickstein, J.; Weiss, E.; Maheswaranathan, N.; and Ganguli, S. 2015. Deep Unsupervised Learning using Nonequilibrium Thermodynamics. In Bach, F.; and Blei, D., eds., *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, 2256–2265. Lille, France: PMLR.

Song, Y.; and Ermon, S. 2019. Generative Modeling by Estimating Gradients of the Data Distribution. In Wallach, H.; Larochelle, H.; Beygelzimer, A.; d'Alché-Buc, F.; Fox, E.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Song, Y.; and Ermon, S. 2020. Improved Techniques for Training Score-Based Generative Models. ArXiv:2006.09011 [cs, stat].

Song, Y.; Sohl-Dickstein, J.; Kingma, D. P.; Kumar, A.; Ermon, S.; and Poole, B. 2021. Score-Based Generative Modeling through Stochastic Differential Equations. In *International Conference on Learning Representations*.

Springel, V.; et al. 2018. First results from the IllustrisTNG simulations: matter and galaxy clustering. *Mon. Not. Roy. Astron. Soc.*, 475(1): 676–698.

Srivatsan, R. A.; Xu, M.; Zevallos, N.; and Choset, H. 2018a. Probabilistic pose estimation using a Bingham distribution-based linear filter. *The International Journal of Robotics Research*, 37(13-14): 1610–1631.

Srivatsan, R. A.; Xu, M.; Zevallos, N.; and Choset, H. 2018b. Probabilistic pose estimation using a Bingham distribution-based linear filter. *The International Journal of Robotics Research*, 37(13-14): 1610–1631.

Thornton, J.; Hutchinson, M.; Mathieu, E.; De Bortoli, V.; Teh, Y. W.; and Doucet, A. 2022. Riemannian Diffusion Schr\"odinger Bridge. ArXiv:2207.03024 [cs, stat].

von Mises, R. 1918. Uber die 'Ganzzahligkeit' der Atomgewicht und verwandte Fragen. *Physikal. Z.*, 19: 490–500.

Zelesko, N.; Moscovich, A.; Kileel, J.; and Singer, A. 2020. Earthmover-Based Manifold Learning for Analyzing Molecular Conformation Spaces. In *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*, 1715–1719.