CSGB CENTRE FOR STOCHASTIC GEOMETRY AND ADVANCED BIOIMAGING

Stefan Sommer and Alex Bronstein

Horizontal Flows and Manifold Stochastics in Geometric Deep Learning

No. 02, February 2020

Horizontal Flows and Manifold Stochastics in Geometric Deep Learning

Stefan Sommer[1] and Alex Bronstein[2]

[1]Department of Computer Science, University of Copenhagen, Denmark.
Email: sommer@di.ku.dk
[2]Computer Science, Technion – Israel Institute of Technology, Israel

Abstract

We introduce two constructions in geometric deep learning for (1) transporting orientation-dependent convolutional filters over a manifold in a continuous way and thereby defining a convolution operator that naturally incorporates the rotational effect of holonomy; and (2) allowing efficient evaluation of manifold convolution layers by sampling manifold valued random variables that center around a weighted Brownian motion maximum likelihood mean. Both methods are inspired by stochastics on manifolds and geometric statistics, and provide examples of how stochastic methods – here horizontal frame bundle flows and non-linear bridge sampling schemes, can be used in geometric deep learning. We outline the theoretical foundation of the two methods, discuss their relation to Euclidean deep networks and existing methodology in geometric deep learning, and establish important properties of the proposed constructions.

Keywords: geometric deep learning, stochastic analysis on manifolds, geometric statistics, frame bundle, curvature, bridge sampling.

1 Introduction

Geometric deep learning [3] concerns the generalization of deep neural network methodology to geometric domains. Focusing on convolutional networks, the complexity in such a generalization appears both in the case where the *domain* of the input signal is non-Euclidean, e.g. a manifold or a graph, and in the case where the *target* of the neural network has geometric structure. A major difficulty in the first case is the fact that translation invariance of the Euclidean convolution operator does not have a direct manifold equivalent: The topology of the geometric space often prevents a continuous transport of the orientation of a filter, and the holonomy of a curved manifold prevents a notion of parallel translation that is independent of the path between points. In particular, parallel translation along minimizing geodesics is not continuous when moving points across the cut locus. In the second case, the weighted Fréchet mean has been proposed as a generalization of the Euclidean convolution to produce manifold valued output. Here, a practical

concern is the computational complexity involved in computing the Fréchet mean on general manifolds.

In this paper, we derive two constructions that seek to provide new perspectives on the above challenges. First, we build on the idea of orientation functions [24] and the use of gauges [8] to show how curvature affects orientations as they are transported backwards through the layers of a multilayer network. The result is a time-discrete horizontal flow in the bundle OM of orthonormal frames of the tangent bundle TM. In relation to gauge equivariant networks [8], the focus here is on the coupling between *transport* of directions and curvature as opposed to equivariance of the convolution operation to gauge transformations. Furthermore, we use the frame bundle and a connection to show how a notion of global parallel transport that circumvents the complexities of nontrivial topology and curvature can be constructed. The idea builds on the Eells-Elworthy-Malliavin construction of Brownian motion [10] that uses horizontal frame bundle flows to construct the Brownian motion on nonlinear manifolds. In the frame bundle, the process results in a distribution of orientations over each point of M, and we use such distributions to construct a convolution operator that transports filters globally over the manifold. The construction is geometrically natural in avoiding linearization to a single tangent space. We build on this idea to construct multilayer convolution using the anti-development of the Brownian motion, resulting in convolution that is both equivariant to frame (gauge) changes, and has a smooth integrand when M is analytic.

Secondly, we combine convolution using the weighted Fréchet mean [5] with the notion of Brownian motion maximum likelihood means on manifolds: center points of a Brownian motion that maximizes the likelihood of a set of manifold valued data. While the weighted Fréchet mean (wFM) can be efficiently computed on manifolds when closed form expressions for geodesics is available, it is computationally more demanding to compute it on general manifolds. We generalize the maximum likelihood mean (mlM) to a weighted maximum likelihood mean (wmlM), and subsequently employ methods from stochastic bridge sampling to *sample* from a distribution centered at the wmlM. This removes the need for expensive iterative optimization. We briefly relate the inherent stochasticity in the construction to other stochastic neural network models.

The paper is divided in two parts targeting the two situations: manifold domain and manifold target, respectively. In each part, we present the background from the geometric deep learning side, connect this with the relevant theory from fiber bundle geometry and stochastics, present the proposed constructions, and prove important properties.

The aim of the paper is to introduce methods from fibre bundle geometry and stochastics on manifolds to the geometric deep learning community from a theoretical viewpoint. We leave actual experimental validation of the methodology to future work. While we focus on continuous manifold geometries, the methods are applicable as well to discrete geometries using discrete connections and parallel transport.

2 Orientations of Filters and Horizontal Frame Bundle Flows

We here aim to show how curvature couples with the change of orientations happening when directions are transported backwards through a multilayer network on the evaluation of the last layer. Subsequently, we use this approach to derive a continuous transport of orientations globally over the nonlinear domain. This allows, for example, to avoid max-pooling over directions before a fully connected final layer that combines information from distant points of the space. The overall aim is to show how the frame bundle FM, the subbundle of orthonormal frames OM, and horizontal flows in the tangent bundle TFM provide a structured way to account for the change of orientations caused by the holonomy of the manifold.

We start with a brief outline of strategies for generalizing convolution to manifold domains, focusing in particular on the directional functions as introduced in [24] and fiber bundles as used in the gauge equivariant networks [8].

2.1 Background

Let M be a d-dimensional Riemannian manifold and $f : M \to \mathbb{R}^{d_{\text{in}}}$ a vector-valued function, e.g. an single channel image ($d_{\text{in}} = 1$) or an RGB image ($d_{\text{in}} = 3$). If M is Euclidean, i.e. $M = \mathbb{R}^d$, each layer in a convolutional network applies the Euclidean convolution $k * f(x) = \int_{\mathbb{R}^d} k(-v)f(x+v)dv$ using a kernel $k : \mathbb{R}^d \to \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$ to give a d_{out}-dimensional output. This is followed by composition with a non-linearity on each component. Discretizing the convolution spatially gives the output as $y(x) = \sum_{i,j} k(-i,-j)f(x + (i,j))$ when $d = 2$ and the sum over i and j runs over the support of the kernel. The kernel k is then specified by a finite set of entries referred to as weights. The linearity of the convolution operation gives rise to the view of each layer as a tensor on functions $M \to \mathbb{R}^{d_{\text{in}}}$. When the convolution appears in the l-th layer of a multilayer network, each component y^n of the vector-valued output can be regarded as a result of a tensor convolution

$$y^n = k_1^n * f^1 + \cdots + k_m^n * f^m \tag{2.1}$$

using a set of kernels k_m^n.

When M is a nonlinear manifold, approaches for generalizing convolution includes spectral methods [4], and techniques using the group structure when M is a Lie group or a homogeneous space [7]. [22], building on [20] and [2], defines the convolution operator using *pseudo-coordinates*, a family of local charts ϕ_x, $x \in M$ that by mapping each point in a neighborhood $U_x \subset M$ to $\mathbb{R}^d$ allows the notion of *patch-operator* to be defined as $D_j f = \int_{U_x} w_j(\phi_x(y))f(y)dy$. The patch operator is subsequently matched to a template to give the final generalized convolution. Particularly, the patch operator can be chosen to be rotationally invariant, e.g. using local geodesic polar coordinates, thus removing ambiguity in the orientation of the chart. However, this significanty restricts the wealth of kernels that can be used in the network.

The pseudo-coordinates in [2] allows rotationally non-invariant kernels by aligning orientations with respect to the directions of maximal curvature direction. However, handling the ambiguity of rotations is not solved entirely in this way because maximal curvature direction may not be defined (e.g. on constant curvature spaces such as spheres); curvature is a local notion which can imply rapid shifts in directions over short distances; topology constrains the set of non-vanishing continuous vector fields (e.g., the hairy-ball theorem on spheres) and so a continuous set of orientations cannot generally be found on topologically non-trivial spaces.

To handle the lack of global orientations on surfaces ($d = 2$), [24] proposes to convolve with functions $f : M \times TM$ that in the second argument take a tangent vector representing a direction. This vector is parallel transported over M along minimizing geodesics resulting in the convolution $k * f(x, v) = \int_{\mathbb{R}^d} k(v) f(\overline{\mathrm{Exp}}_x(uv))dv$ where the map $\overline{\mathrm{Exp}}_x$ is the Riemannian exponential map $\mathrm{Exp}_x : T_xM \to M$ combined with parallel translation of the vector v to provide directional information for the evaluation of f. The map u is a frame as described below. The use of directional functions implies that directions are propagated between layers in a consistent way. Parallel transport is also used to define convolution in [27].

Gauge equivariant networks [8] provide a related approach to handle directional ambiguity. A gauge for the tangent bundle TM is a map $u : U \times \mathbb{R}^d \to TM$ that for each x in an open set $U \subset M$ gives an invertible linear map $u_x : \mathbb{R}^d \to T_xM$. Equivalently, a gauge is a local section of the bundle of ordered bases of the manifold, the frame bundle $\pi : FM \to M$. Let $\pi_{\mathrm{in}} : N_{\mathrm{in}} \to M, \pi_{\mathrm{out}} : N_{\mathrm{out}} \to M$ be two vector bundles over M. A gauge equivariant convolution takes as input a section $f : M \to N_{\mathrm{in}}$ of N_{in} and outputs a section of N_{out} given by $k * f(x) = \int k(v)\rho_{x \leftarrow \mathrm{Exp}_x(u_x v)} f(\mathrm{Exp}_x(u_x v))dv$ where $P_{x \leftarrow p} : \pi_{\mathrm{in}}^{-1}(p) \to \pi_{\mathrm{in}}^{-1}(x)$ is a transport operation in N_{in}. Here, gauges enter in the kernel as $k(v) = u_{x,\mathrm{out}} k(v) u_{x,\mathrm{in}}^{-1}$ where $k(v) \in \mathbb{R}^{d_{\mathrm{in}}, d_{\mathrm{out}}}$ and $u_{x,\mathrm{in}}$, $u_{x,\mathrm{out}}$ are frames for $N_{\mathrm{in}}, N_{\mathrm{out}}$, respectively. The convolution can be shown to be independent of the choice of gauges if k satisfies an invariance condition [8] dependent on the structure group, see also [17]. Particularly relevant is equivariance to the rotation group $\mathrm{SO}(d)$, equivalently choices of orthonormal frames in the bundle OM described below. In this case, gauge equivariance for scalar valued functions is equivalent to rotational invariance.

The transport operation $P_{x \leftarrow p}$ is parallel along a unique minimizing geodesic, either by parallel translating each vector of $u_{x,\mathrm{in}}$ to p or by using a connection on the bundle N_{in}. For this reason, we write $P^{-1}_{\gamma(x, u_x v)}$ for $P_{x \leftarrow \mathrm{Exp}_x(u_x v)}$ below, where $\gamma(x, w)$ denotes the geodesic starting at x with derivative $\dot{\gamma}(0) = w$. The uniqueness of the geodesic between x and p is only ensured away from the cut locus of the manifold. The kernel k must therefore have limited support in order for the operation to be well-defined.

2.2 Directional Functions

A natural generalization to higher dimensions of the convolution of directional functions on surfaces in [24] is to define convolution on functions $f : OM \to \mathbb{R}$ where OM is the orthonormal frame bundle, i.e. the subbundle of the frame bundle FM

consisting of *orthonormal* frames. Then convolution can be defined as

$$k * f(u) = \int_{\mathbb{R}^d} k(-v) f(P_{\gamma(x,uv)}(u)) dv, \;\; x = \pi(u) \tag{2.2}$$

Note that the frame bundle element u is used to map the $\mathbb{R}^d$ vector v to the vector uv in T_xM to give the direction of the geodesic $\gamma(x, uv)$. The frame u is then parallel transported along this geodesic to enable evaluation of f. Here, we focus on real valued functions and kernels $k : \mathbb{R}^d \to \mathbb{R}$ though the construction and the additional convolutions defined below extend to multiple output channels or bundle valued outputs.

Remark 2.1. Note the difference between this construction and the gauge equivariant case: The input functions in the latter case have no directional input information; instead they are equivariant to gauge changes, e.g., the action of $\mathrm{SO}(d)$. However, for $f : OM \to \mathbb{R}$ and $x \in M$, we can view the restriction $f|_{\pi|_{OM}^{-1}(x)}$ to the fiber over x as an element of the bundle $N = \{x \in M |\; h : \pi|_{OM}^{-1}(x) \to \mathbb{R}\}$ of functions on the fibers $\pi|_{OM}^{-1}(x)$. The group $\mathrm{SO}(d)$ acts on N on the right by $g.h(u) = h(g.u)$, and the convolution (2.2) can be seen as a gauge equivariant network with $N_{\text{in}} = N_{\text{out}} = N$. In fact, in this case, equivariance implies that any convolution output is a function on the fibers because of the dependence on the frame/gauge. Orientation functions can be seen as continuous analogues of the discrete rotations used in [8].

In the convolution (2.2), it is important to note that directional information propagates *backwards* through a composition of layers: As illustrated in Figure 1, let $f_1 = k_1 * f$ and $f_2 = k_2 * f_1$. Then in the convolution to produce $f_2(u)$, u is parallel transported along geodesics from $x = \pi(u)$ in order to evaluate f_1. The frames $P_{\gamma(x,uv)}(u)$ are then in turn parallel transported a second time before evaluating f. Because of the path dependence of parallel transport, this is in general not equal to parallel transporting only once if evaluating a filter $(k_2 * k_1) * f$ with $k_2 * k_1$ denoting the standard Euclidean convolution. We show below how this difference is related to the curvature of M.

The convolutions defined in [24], [8] and (2.2) above implicitly construct a gauge in the evaluation of the integral in the convolution because the parallel transport $P_{\gamma(x,uv)}(u)$ gives a local section of OM. This is a specific choice of gauge, and a different choice would result in different results of the convolution. In particular, a different choice of paths along which u is parallel transported would have this effect (see also discussion in [6]). Below, we will embrace this by defining a measure on such paths and integrating out the effects of the difference in parallel transport. As noted above, k is often assumed to have limited support implying that the choice of paths may not have a great effect. However, this may not be the case when the output features of distant points are compared in fully connected layers appearing as the last layers of a multilayer network. Currently, max-pooling over directions is often applied before such a layer [8]. Below, we construct a principled way to integrate rotations without such a pooling.

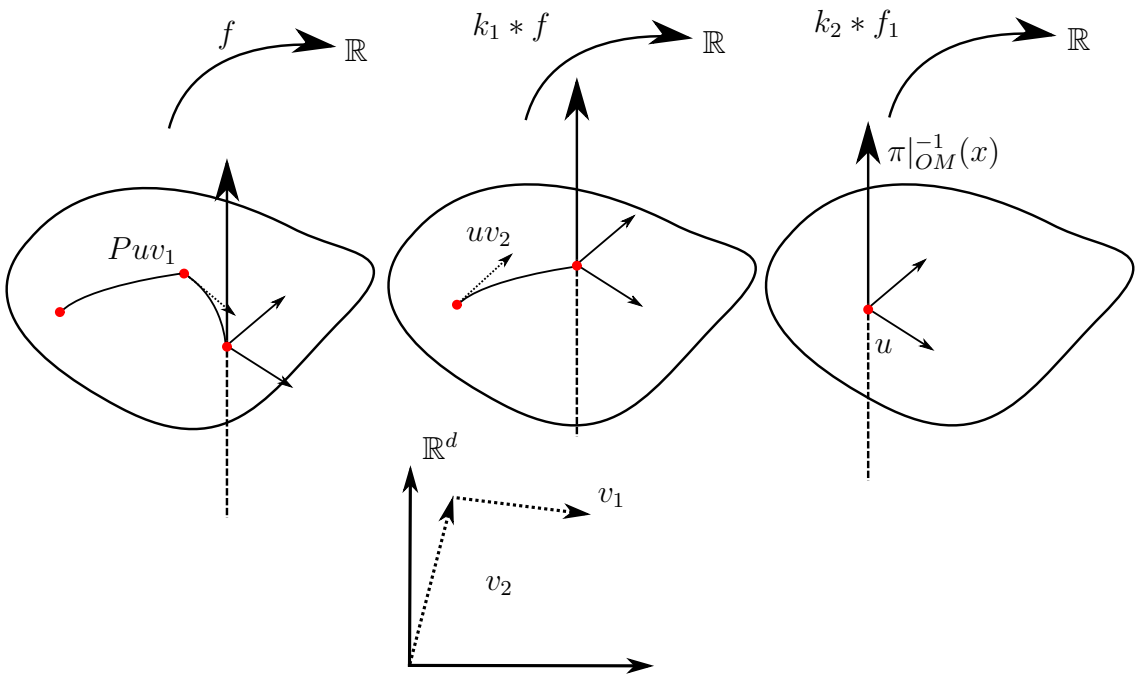

Figure 1: Directional information captured in the frame $u \in OM$ which is an element of the fiber $\pi_{OM}^{-1}(x)$ (illustrated by vertical arrow in rightmost sketch) over a point $x \in M$ (red dot). In the convolution (2.2), u is parallel transported along geodesics with initial direction uv_2 for vector $v_2 \in \mathbb{R}^d$ (center). When composing convolutions, the frame $P_{\gamma(x,uv_2)}u$ at $\gamma(x, uv_2)$ is transported along a second geodesic with direction $P_{\gamma(x,uv_2)}uv_1$ (left). The directional information in u is thus transported *backwards* through the layers of a multi-player network. The curvature implies that this transport is path dependent: A different choice of path would yield a different transport.

2.3 Parallel Transport and Horizontality

To further highlight the geometric setting for the use of parallel transport in the convolutions above, we here outline the necessary fiber bundle geometry leading to the view of parallel transport as a horizontal OM flow. Figure 2 shows the maps between the bundles used below. Further details on frame bundles as used here can for example be found in the books [15, 16]. The frame bundle has been used in the context of geometric deep learning before, e.g. in [8]. Here we use the frame bundle as well, however, we focus on flows in the frame bundle, i.e. the *transport* of frames.

As mentioned previously, OM is a fiber bundle, a subbundle of the frame bundle $\pi : FM \to M$. The map π attaches the base point $x = \pi(u)$ in M to each element $u \in FM$. $u = (u_1, \ldots, u_d)$ is a frame, an ordered set of tangent vectors $u_i \in T_{\pi(u)}M$ making up a basis for $T_{\pi(u)}M$. For elements of OM, the basis is orthonormal with respect to a Riemannian metric g: $g(u_i, u_j) = \delta_{ij}$, $i, j = 1, \ldots, d$. Each frame $u \in FM$ provides an invertible linear map $\mathbb{R}^d \to T_xM : v \to \sum_{i=1}^d u_i v^i$, and FM and OM can therefore be viewed as the principal fiber bundles $\mathrm{GL}(\mathbb{R}^d, TM)$, and $\mathrm{O}(\mathbb{R}^d, TM)$. $\mathrm{GL}(d)$ naturally acts on FM on the right by $g.u \mapsto u \circ g$. The structure group is therefore $\mathrm{GL}(d)$. For OM, the structure group is the subgroup $\mathrm{O}(d)$.

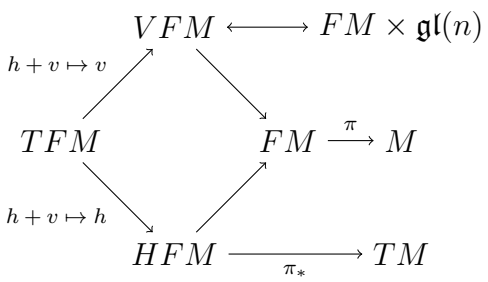

Figure 2: Relations between the manifold, frame bundle, the horizontal distribution HFM, and the vertical bundle VFM. The connection $\mathcal{C}$ on FM provides the splitting $TFM = HFM \oplus VFM$. The restrictions $\pi_*|_{H_u M}$ of the push-forward of the projection map $\pi : FM \to M$ are invertible maps $H_u M \to T_{\pi(u)} M$ with inverse h_u, the horizontal lift. The vertical bundle VFM is isomorphic to the trivial bundle $FM \times \mathfrak{gl}(n)$.

A connection ∇ on M, e.g., the Levi-Civita connection, lifts to a fiber bundle connection $\mathcal{C}$ on FM: A path $u(t) \in FM$ has zero acceleration if and only if each basis vector $u_i(t)$ is parallel transported on M. The connection $\mathcal{C}$ provides a split of the tangent bundle TFM into *vertical* and *horizontal* components: The vertical component VFM is the subbundle $\{v \in TFM : \pi_*(v) = 0\}$, i.e., derivatives of paths $u(t)$ satisfying $u(t) = x$ for all t. That is, the base point is fixed and only the frame changes. The horizontal subbundle HFM consists of derivatives of zero-acceleration, paths in FM along which each basis vector $u_i(t)$ is parallel transported along the path $\pi(u(t))$ in M. Such paths are called *horizontal*. The connection $\mathcal{C}$ is then explicitly a projection $TFM \to VFM$, and, using this, TFM can be split into the direct sum $TFM = VFM \oplus HFM$. HFM and VFM being subsets of TFM are also denoted the horizontal and vertical distributions, respectively.

Because of this decomposition of TFM, any vector $v \in T_x M$ can be lifted to a unique vector in $H_u FM$, $\pi(u) = x$. This operation written as $h_u : T_{\pi(u)} M \to H_u FM$ is denoted as the *horizontal lift* of v. In particular, the basis vectors $u_1, \ldots, u_d$ can be lifted to vectors $H_i(u) := h_u(u_i)$, $i = 1, \ldots, d$. This gives the set of *horizontal vector fields* on FM. Importantly, the fields H_i are globally defined, smooth, and, for each u, they provide a basis for $H_u FM$. We now have the geometric setup to describe the parallel transport as used in (2.2) and [24, 8] as a flow in OM:

Lemma 2.2. *Let $v \in \mathbb{R}^d$, $u \in OM$ and $x = \pi(u)$. The transport $u(t) = P_{\gamma(x,tuv)}(u)$ is an integral curve of a horizontal flow in OM.*

Proof. For $v \in \mathbb{R}^d$, $h_u(uv) = \sum_{i=1}^d H_i(u) v^i$ is a vector field on FM. Let $\Phi : FM \times \mathbb{R} \to FM$ be the unique flow satisfying $\partial_t \Phi(u) = h_u(uv)$, and set $u(t) := \Phi(u, t)$. Then $u(t)$ is horizontal because $h_u(uv) \in HFM$, and thus, for each $i = 1, \ldots, d$, $u_i(t)$ is parallel transported along $\pi(u(t))$. In particular, $uv = \sum_{i=1} u_i v^i$ is parallel transported along $\pi(u(t))$, and because $uv = \partial_t \pi(u(t))$, $\pi(u(t))$ is the geodesic $\gamma(x, uv)$. $u(t)$ is orthonormal for all t because $u \in OM$ and the distribution HFM is tangent to OM. □

While the lemma only gives an expression for the parallel transport $P_{\gamma(x,uv)}(u)$ in the language of horizontal flows, it provides the basis for understanding the coupling between layers and the stochastic horizontal flows described below.

2.4 Horizontality, Curvature and Composition of Layers

Associativity $(k_2 * k_1) * f = k_2 * (k_1 * f)$ of the Euclidean convolution is a consequence of its translation invariance. Parallel transport of frames along geodesics implies translation invariance along those geodesics, but the path dependence rules out translation invariance between points when the path is not specified. For one layer filters with limited support, one can reasonably restrict to transport along geodesics as above. But with multiple layers, the integral in the convolution is evaluated along compositions of geodesics giving curves that are only piecewise geodesic as illustrated in Figure 1. This again implies a rotation of filters. We here connect this fact to horizontal OM flows and curvature.

The following statement which comes from an application of Taylor's theorem on OM, uses parallel transport and OM to express the lack of associativity and commutativity of the convolution operation directly in terms of the curvature of M. We use the bracket $[h_u(v), h_u(w)]f = h_u(v)h_u(w)f - h_u(w)h_u(v)f$ for $v, w \in T_xM$ which measures the lack of commutativity of derivatives by horizontal vector fields. The bracket is directly linked to the curvature tensor R of M by the relation

$$R(v, u) = -\mathcal{C}([h_u(v), h_u(w)]) \ , \tag{2.3}$$

i.e., the curvature measures the vertical component of the bracket between horizontal vector fields.

Theorem 2.3. *Let k_1, k_2 be kernels with $\mathrm{supp}(k_i) \subseteq B_r(0)$, and $f \in C^3(OM, \mathbb{R})$. Then*

$$\begin{aligned} &k_2 * (k_1 * f) - (k_2 * k_1) * f \\ &\quad = \int_{\mathbb{R}^d} \int_{\mathbb{R}^d} k_2(-v_2) k_1(-v_1) h_u(v_2) h_u(v_1) f dv_1 dv_2 + o(r^{d+1}) \end{aligned} \tag{2.4}$$

and

$$\begin{aligned} &k_2 * (k_1 * f) - k_1 * (k_2 * f) \\ &= \int_{\mathbb{R}^d} \int_{\mathbb{R}^d} k_2(-v_2) k_1(-v_1) [h_u(v_2), h_u(v_1)] f dv_1 dv_2 + o(r^{d+1}). \end{aligned} \tag{2.5}$$

In particular, the vertical part of the non-commutativity in (2.5) *is a function of the curvature tensor $R(v_2, v_1)$.*

Proof. Let $f^v : OM \to \mathbb{R}$ be the map $u \mapsto f(P_{\gamma(\pi(u), uv)} u)$ (the map in the integrand of (2.2)). By Taylor's theorem and Lemma 2.2, $f^v(u) = f(u) + h_u(v)f + o(\|v\|)$. Applying Taylor's theorem again, we get $(f^{v_1})^{v_2}(u) = f(u) + h_u(v_1)f + h_u(v_2)f + h_u(v_2)h_u(v_1)f + o(\|v_1\|, \|v_2\|)$. Then, using the regular Euclidean convolution $k_1 * k_2$,

$$\begin{aligned} &k_2 * (k_1 * f)(u) - (k_2 * k_1) * f(u) \\ &\quad = \int_{\mathbb{R}^{2d}} k_2(-v_2) k_1(-v_1) ((f^{v_1})^{v_2}(u) - f^{v_1 + v_2}(u)) d(v_1, v_2) \\ &\quad = \int_{\mathbb{R}^{2d}} k_2(-v_2) k_1(-v_1) h_u(v_2) h_u(v_1) f) d(v_1, v_2) + o(r^{d+1}) \ . \end{aligned}$$

The commutativity relation (2.5) results from using (2.4) and commutativity of the Euclidean convolution. □

The result makes explicit the relation between non-commutativity and non-associativity of convolution kernels when using parallel transport and the curvature of the manifold, equivalently non-integrability of the horizontal distribution HFM as seen by the brackets $[h_u(v_1), h_u(v_2)]$ being nonzero. One can also view the use of the Riemannian exponential map and parallel transport along geodesics as a linearization of the manifold [32]. The lack of commutativity is a reflection of the fact that such linearizations generally do not lift to subspaces of the frame bundle.

2.5 Convolution with Horizontally Distributed Orientations

When applying multiple convolutions, the consecutive application of parallel transport is related to time-parameterized flows in stochastic analysis on manifolds. We first describe the Riemannian Brownian motion as an example of such a flow and use it to distribute orientations along multiple paths between pairs of points. Subsequently, we show how compositions of many layers can be seen as a time-discretized OM flow.

Let $(\Omega, (\mathcal{F}_t)_{t\geq 0}, P)$ be a standard probability space with filtration $(\mathcal{F}_t)_{t\geq 0}$. The Riemannian Brownian motion is a stochastic process X_t, i.e. a time-indexed sequence of $\mathcal{F}_t$ measurable, M-valued random variables, that has density $p_t(\cdot; x)$ where x denotes the starting point of the process, i.e. $X_0 = x$. p_t is also denoted as the heat kernel, and it satisfies the heat equation $\partial_t p_t = -\frac{1}{2}\Delta p_t$ where Δ is the Laplace-Beltrami operator of M. p_t is smooth for all $t > 0$, and non-zero if M is connected. The heat flow has been used extensively in geometric deep learning [4]. Here we focus on its relation to parallel transport.

The construction of the Riemannian Brownian motion is non-trivial due to the global nature of the process but the local nature of charts, see, e.g., [11]. One approach is the Eells-Elworthy-Malliavin construction [10] that avoids the use of charts by mapping horizontal OM flows to M: An $\mathbb{R}^d$-valued Euclidean Brownian motion W_t is mapped to an OM-valued stochastic process U_t by the SDE $dU_t = \sum_{i=1}^{d} H_i \circ_{\mathcal{S}} dW_t^i$ where $\circ_{\mathcal{S}}$ denotes Stratonovich integration, see, e.g., [15] for details. The starting point U_0 is one point in $u \in OM$. We make the dependence on the starting point explicit by writing U_t^u. By mapping U_t^u to the manifold, the resulting process $X_t^x = \pi(U_t^u)$ is a Brownian motion with starting point $x = \pi(u)$. Figure 3 illustrates the relation between W_t, U_t and X_t. Long-time existence of the Brownian motion can be proven under mild assumptions on M (e.g. compactness is sufficient).

For each t, U_t is an OM-valued random variable and X_t is an M-valued random variable. The distribution corresponding to X_t has density $p_t(\cdot; x)$. U_t may also have a smooth density on OM, however, this depends on the curvature: U_t may at time t hit a fiber $\pi|_{OM}^{-1}(y)$, $y \in M$ along many different paths on M, not just geodesics. Each such path will have its own parallel transport, and which point in the fiber is hit depends on the path. The difference will be a shift of orientation, i.e., a gauge transformation. U_t thus gives a distribution of orientations for each fiber $\pi|_{OM}^{-1}(y)$. For flat manifolds, parallel transport is path independent, and U_t is supported on a d-dimensional submanifold of OM. Conversely, on curved manifolds with holonomy group $\mathrm{SO}(d)$, all rotations appear, U_t will be non-zero on all of OM, and it will have a smooth, positive density.

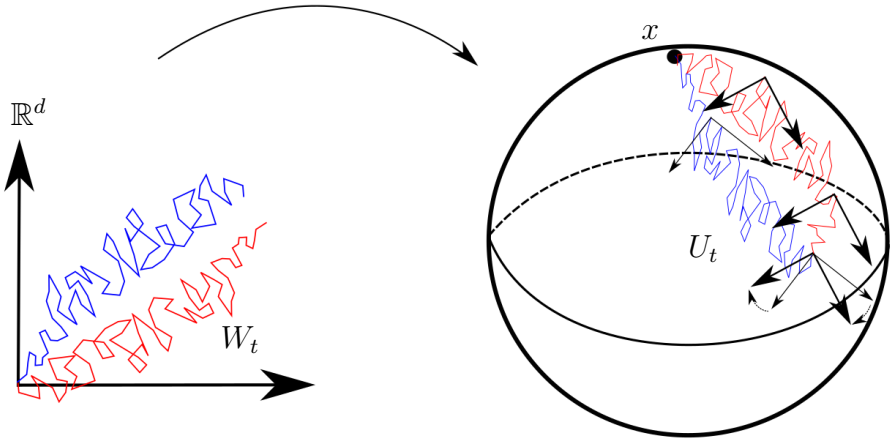

Figure 3: The relation between the Euclidean $\mathbb{R}^d$-valued Brownian motion W_t, the OM process U_t that carries the frame by parallel transporting along the stochastic paths $X_t = \pi(U_t)$ on M. W_t is mapped to U_t and X_t by *development*. The reverse mapping is denoted *anti-development*. Two sample paths $X_t(\omega_1)$, $X_t(\omega_2)$ (blue/red) ending at the same point in M ($X_T(\omega_1) = X_T(\omega_2)$) need not end at the same point when anti-developed to $\mathbb{R}^d$. The curvature implies that $U_T(\omega_1)$, $U_T(\omega_2)$ may hit different points in the fiber over the endpoint. The difference is a rotation (or gauge transformation).

While for the convolution (2.2) we only used geodesics from a point x in the parallel transport, the OM-flow defines a probability measure $P_{U_t^u}$ on stochastic paths from x. We can use this to define a convolution that takes any $P_{U_t^u}$-measurable path into account:

$$k *_{U_T^u, dv} f(u) = \int_{\mathbb{R}^d} \int_{\pi^{-1}(\mathrm{Exp}_x(uv))} k(-v) f(U_T^u) P(dU_T^u) dv \tag{2.6}$$

Note that the definition integrates over each fiber $\pi^{-1}(\mathrm{Exp}_x(uv))$ in OM with the distribution on each fiber being a result of the OM flow U_t^u at a fixed time $T > 0$, see Figure 4. The notation $*_{U_T, dv}$ makes the dependence on the measures $P(dU_T^u)$ and dv in the integration explicit.

For kernels with limited support, minimizing geodesics are in practice unique, and parallel transport can reasonably be performed along minimizing geodesics. This is generally the case for convolutional layers. However, the last layers of a network can be fully connected, and one thus cannot limit the support of the kernel. Instead, pooling over rotations can be performed to remove the rotational ambiguity of non-unique geodesics. In contrast, the convolution (2.6) allows a fully connected layer to include information from the entire manifold while handling rotations in a principled way. In particular, the orientation distribution is continuous as a function of u when M is analytic as discussed below.

2.6 Multilayer Convolution as Stochastic Flow

We now aim to improve the convolution (2.6) to express it in stochastic terms, and to give it a natural behaviour when composing layers. For this, we need the notion

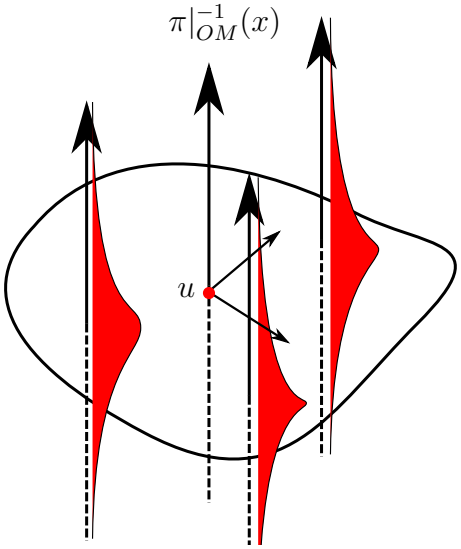

Figure 4: The convolution (2.6) uses the OM distribution U_T^u, $T > 0$ that in each fiber has a distribution of frames, i.e. a distribution of orientations (red fiber density illustration). With zero curvature, parallel transport is path independent, and the distribution will be singular supported at a single element of each fiber. With curvature, the fiber distributions widen, possibly filling the fibers. For analytic manifolds, U_T^u has smooth, positive density for any $T > 0$ on a submanifold of OM.

of stochastic development and anti-development that expresses the relation between the processes W_t, U_t, and X_t. First, let us rewrite (2.6) to avoid the split between the fiber integration $P(dU_T^u)$ and the Euclidean integration dv. We do that using the distribution U_T^u directly:

$$\begin{aligned} k *_{U_T^u,\mathrm{Log}} f(u) &= \int k(-u^{-1}\operatorname{Log}_x(\pi(U_T^u)))f(U_T^u)P(dU_T^u) \\ &= \mathrm{E}[k(-u^{-1}\operatorname{Log}_x(\pi(U_T^u)))f(U_T^u)] \end{aligned} \tag{2.7}$$

where Log_x denotes the local inverse of the Exp_x, and E the expectation with respect to the law of U_T^u. Log_x here provides pseudo-coordinates: Because the integration is now over OM, we need to map to $\mathbb{R}^d$ to evaluate the kernel k. This again makes the integrand discontinuous because of the local nature of Log_x (the logarithm is discontinuous when crossing the cut locus). It turns out that this deficiency can be removed.

Recall the connection between the $\mathbb{R}^d$-valued Brownian motion W_t in the Eells-Elworthy-Malliavin construction and the M-valued process $X_t = \pi(U_t)$. X_t is denoted the *stochastic development* of W_t, since X_t is developed, or rolled-out, over M following the stochastic increments dW_t for each time t using the current values of the process U_t to map from $\mathbb{R}^d$ to $T_{X_t}M$. The reverse is also true: Any M-valued semimartingale can be *anti-developed* to a semi-martingale on $\mathbb{R}^d$. Figure 3 illustrates the relation between sample paths $W_t(\omega)$ and the developments $U_t(\omega)$. Using this relation directly, we can define a convolution as

$$k *_{W_T} f(u) = \int k(-W_T)f(U_T^u)P(dW_t) = \mathrm{E}[k(-W_T)f(U_T^u)] \tag{2.8}$$

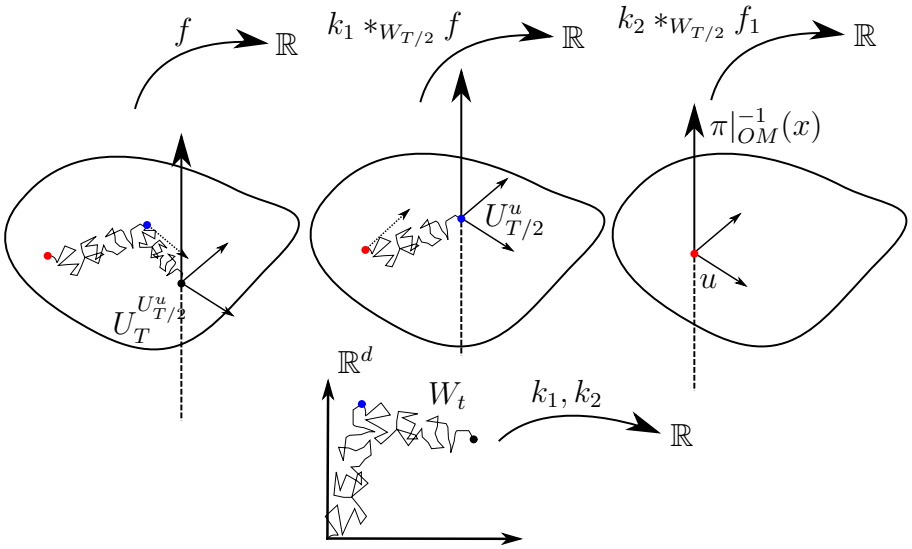

Figure 5: The convolution (2.8) applies kernels, here k_1, k_2, on the antidevelopment W_t whereas f is applied on the OM process U^u_T. Compare with Figure 1.

where the mapping from W_t to U^u_t by development is used, and the expectation on the right-hand side is with respect to the law of the Brownian motion W_t. Note that k is evaluated on W_T for a fixed T. This Euclidean random variable is in fact normally distributed. However, f is evaluated on the OM-valued random variable U^u_T that automatically includes directional information. In comparison with (2.7), Log_x is not used in (2.8).

The convolution depends on the Brownian motion W_t up until the evaluation time $T > 0$. Varying T will change the distribution of orientations over M: For T large, all orientations will diffuse to be equally probable; in the limit $T \to 0$, the convolution (2.2) is recovered because the U^u_T measure concentrates around the points in each fiber that corresponds to parallel transport along geodesics from x (see small-time asymptotic limit results in, e.g., [15] and [31]).

Remark 2.4. The OM endpoint U^u_T is dependent on the entire path U^u_t, $t \in [0, T]$: Let ω^1, ω^2 be two elements of Ω such that $W_T(\omega^1) = W_T(\omega^1)$. Then $U^u_T(\omega^1)$ does not necessarily equal $U^u_T(\omega^2)$. This is a consequence of curvature and reflects that development is a map from the path space $W([0, T], \mathbb{R}^d)$ to the path space $W([0, T], OM)$, i.e. the endpoint $U^u_T(\omega)$ is dependent on the entire path $W_t(\omega)$. The path spaces are Wiener spaces of continuous paths on $[0, T]$.

Brownian motion or, equivalently, the heat flow is a semi-group which is often expressed in terms of the density: $p_{t+s}(y; x) = \int_M p_s(y; z) p_t(z; x) dz$. In other words, we can obtain W_{t+s}, U^u_{t+s}, and X^x_{t+s} by starting the stochastic processes at 0, u, and x respectively, running the process to time t, and then restart the processes at W_t, U^u_t and X^x_t to obtain $W_{t+s} = W^{W_t}_s$, $U^u_{t+s} = U^{U^u_t}_s$, and $X^x_{t+s} = X^{X^x_t}_s$. We can use this and development to express compositions of convolution as one integral over the Brownian motion with the filters applied at discrete times. To see this, compose

two $W_{T/2}$ convolution layers to get

$$\begin{aligned} & k_2 *_{W_{T/2}} (k_1 *_{W_{T/2}} f)(u) \\ & \quad = \mathrm{E}[k_2(-W_{T/2})(k_1 *_{W_{T/2}} f)(U^u_{T/2})] \\ & \quad = \mathrm{E}[k_2(-W_{T/2})\,\mathrm{E}[k_1(-(W_T - W_{T/2}))f(U_T^{U^u_{T/2}})]] \\ & \quad = \mathrm{E}[k_2(-W_{T/2})k_1(-(W_T - W_{T/2}))f(U^u_T)] \end{aligned} \tag{2.9}$$

using the semigroup property for W_t and U_t. Note that $W_T - W_{T/2}$ is Gaussian distributed with variance equal to $W_{T/2}$. Thus, with n layers and filters $k_1, \dots k_n$,

$$\begin{aligned} & k_n *_{W_{T/n}} (k_{n-1} *_{W_{T/n}} \cdots (k_1 *_{W_{T/n}} f))(u) \\ & \quad = \mathrm{E}[k_n(-W_{T/n})k_{n-1}(-(W_{2T/n} - W_{T/n})) \\ & \qquad \cdots k_1(-(W_T - W_{(n-1)T/n}))f(U^u_T)] \end{aligned}$$

For the evaluation of the output layer at u, the result of the convolution, with f being the input function, the stochastic flow visits the layers evaluating k_n at $t = T/n$ and k_1 at $t = T$. *Forward* time of the processes W_t and U^u_t thus implies *backwards* visits through the layers. All differences $W_{iT/n} - W_{(i-1)T/n}$ are normally distributed in $\mathbb{R}^d$. The base points $\pi(U^u_T)$ in M follows the distribution of a Brownian motion started at $\pi(u)$ evaluated at $t = T$, and orientations are distributed in OM by parallel translating along the stochastic paths $\pi(U_t)$ in M. The effect of the convolution can be seen by comparing Figure 5 with Figure 1.

2.7 Properties

Tensor convolutions: When the convolution appear in the l'th layer of a multiplayer network with multi-dimensional input and output, we can generalize the tensor convolution (2.1) by writing (2.8) in the form

$$\begin{aligned} y^n &= \mathrm{E}[k^n(-W_t)f(U^u_T)] \\ &= \mathrm{E}[k^n_1(-W_t)f^1(U^u_T)] + \cdots + \mathrm{E}[k^n_m(-W_t)f^m(U^u_T)] \end{aligned}$$

with a set of kernels k^n_m being the entries of the kernel tensor k. The linearity of the expectation thus implies that convolution can be extended to tensor convolution similarly to the Euclidean case.

Equivariance: Let $f : OM \to \mathbb{R}$ and $g \in \mathrm{O}(d)$. As mentioned previously, g acts on the right on f by $(g.f)(u) = f(g.u)$ (recall that $\mathrm{GL}(d)$ acts on FM by right composition). Then

$$\begin{aligned} k *_{W_T} (g.f)(u) &= \mathrm{E}[k(-W_T)f(U^u_T \circ g)] = \mathrm{E}[k(-W_T)f(U^{u\circ g}_T)] \\ &= k *_{W^T_t} f(g.u) = g.(k *_{W_T} f)(u) \end{aligned}$$

because the parallel transport in U^u_t acts on u by composition on the left. The horizontal convolution is thus equivariant to the g action on functions $OM \to \mathbb{R}$.

Smoothness: When the finite bracket span of H_uFM, i.e. the span of iterated brackets $[[[H_{i_1}(u), H_{i_2}(u)], H_{i_3}(u)], \ldots, H_{i_r}(u)]$, $r \in \mathbb{N}$, $i_j = 1, \ldots, d$, generates a subspace of T_uOM of constant rank as a function of u, there exists a submanifold of OM on which U_t^u has a smooth, positive density for all t by the Frobenius theorem. In this case, the integrand $f(U_T^u)$ in the convolution inherits the smoothness of f. This is in contrast to the parallel transport of frames along minimizing geodesics where the minimizing geodesics shift discontinuously when crossing the cut locus. The constant rank condition is for instance satisfied for analytic manifolds [21] and homogeneous spaces.

Nonlinearities: With addition of layer-wise nonlinearities ϕ_i, $i = 1, \ldots, n$, the full network takes the form

$$\begin{aligned}
&\phi_n(k_n *_{W_{T/n}} \phi_{n-1}(\cdots \phi_1(k_1 *_{W_{T/n}} f))(u) \\
&\quad = \phi_n(\mathrm{E}[k_n(-W_{T/n})\phi_{n-1}(\mathrm{E}[k_{n-1}(-(W_{2T/n} - W_{T/n})) \\
&\qquad \cdots \phi_1(\mathrm{E}[k_1(-(W_T - W_{(n-1)T/n}))f(U_T^u)])])])
\end{aligned}$$

Spatial pooling and Gaussian weighting: There is an implicit Gaussian weighting in the integral in (2.8) since W_T is normally distributed. This is in contrast to the most standard form of convolution where the integral is taken with respect to the Lebesgue measure on $\mathbb{R}^d$. This can be compensated for in reweighting the kernel, i.e. exchanging $k(x)$ with $k(x)/p_T(x)$ where p_T is the density of the centered normal distribution in $\mathbb{R}^d$ with variance T. The use of the Brownian motion makes the construction related to the diffusion-convolution networks [1], and the anisotropic heat flow used to construct patch operators in [2]. However, the focus here is on distributing orientations in OM as opposed to constructing a density or defining patches on M.

It is common practice to use a form of spatial pooling in convolutional networks. Average pooling is by construction convolution with a box kernel. With stride, it provides a coarsened version of the discretized output function similarly to max-pooling. The Gaussian weighting of the integral gives a similar effect when convolving the result of a convolution with an identity kernel: Letting $k_2(x) = 1$ in (2.9), we get $k_2 *_{W_{T/2}} (k_1 *_{W_{T/2}} f)(u) = \mathrm{E}[k_1(-(W_T - W_{T/2}))f(U_T^u)]$ where it can be seen that f is evaluated at time T of the Brownian motion whereas k_1 is evaluated at $W_T - W_{T/2}$ which has half the variance. This can be seen as a "Gaussian stride": f is evaluated at points having twice the variance as the evaluation points of the kernel thus mimicking the way regular stride doubles the length scale on which the input function is evaluated. In the Euclidean situation, the result can be seen as exchanging the average filter in average pooling with a convolution of the output with a Gaussian kernel of larger width.

2.8 Numerical Implementation

While the heat equation on manifolds is a nonlinear PDE, the heat kernel can be numerically computed efficiently on discretized surfaces. The vector heat method [28] lifts this to transport in the tangent bundle. We expect these methods to be transferable to efficient numerical evaluation of the expectation in (2.8) though an actual

implementation is left to future work. Stochastic horizontal transport, development and Monte Carlo approximation of $p_t(\cdot; x)$ with bridge sampling is implemented in the Theano Geometry framework [18].

Because the convolution (2.8) inherits the smoothness of the input function when M is, e.g., analytic, it is possible to backpropagate through the operation, the actual operation depending on the chosen numerical scheme. Because U_t^u is a horizontal flow, the computation will involve derivatives of the horizontal vector fields, and thus be related to the curvature of M.

2.9 Conclusion

We have outlined fiber bundle geometry that allows the use of parallel transport in geometric deep learning to be viewed as integral curves of horizontal flows, and we used this to explicitly link the lack of associativity and commutativity of convolutions using parallel transport to the curvature of the manifold. Subsequently, we used stochastic horizontal OM flows to distribute orientations globally over M, and we applied development and anti-development to define a new convolution operator that naturally includes orientations in its definition trough the use of frame bundle flows. The construction does not rely on minimizing geodesics and therefore removes the orientation ambiguity when using kernel with large support.

3 Sampling Means for Manifold Valued Convolutional Filters

We now switch focus and consider the situation of a manifold valued convolutional filter, i.e. $k * f$ takes values in M. The complexity here lies in the fact that there is no direct way to enforce the value of an integral to take values in a manifold. This problem has been the focus of intensive interest in geometric statistics, the statistical analysis of data in geometric spaces: Fréchet defined in [12] a generalization of the Euclidean expected valued as the Fréchet mean (FM), and this and related notions of manifold means have been treated in numerous works. Relevant for the present context is the introduction of weights and the weighted Fréchet mean (wFM) which in [5] is used to define a generalization of the Euclidean convolution that takes values in a manifold.

Because both the Fréchet mean and the weighted Fréchet mean are posed as optimization problems – minimizers of the (weighted) variance, they are typically expensive to compute, which is a major obstacle in deep learning applications. This issue can be handled in spaces where geodesics have closed form solution using an inductive estimator [5]. Here, we take a different view on the estimation problem and propose a method for *sampling* from a distribution centered around weighted means, thus removing the need for optimization steps to find shortest geodesics and a minimizer of the expected variance. For this, we introduce the *weighted Brownian motion maximum likelihood mean*, a version of mean value that is defined from a likelihood principle in contrast to the non-parametric definition of the Fréchet mean. We develop a novel sampling scheme in the n-fold product manifold M^k for

k points by conditioning a stochastic process to hit the diagonal of M^k, identify the distribution of the resulting random variable, and relate the introduced stochasticity to other stochastic neural networks models.

3.1 Background

Euclidean convolution can be written $k * f(x) = \mathrm{E}[k(x-z)f(z)]$ with expectation with respect to the Lebesgue measure. Assuming $\mathrm{E}[k] = 1$, the result can equivalently be expressed as $k*f(x) = \operatorname{argmin}_{y\in\mathbb{R}} \mathrm{E}[k(x-z)\|y-f(z)\|^2]$ where $\|\cdot\|^2$ denotes the squared Euclidean norm by differentiating at optimal y, see e.g. [14]. While the expected value does not have a manifold equivalent, the Riemannian generalization $\operatorname{argmin}_{y\in\mathcal{M}} \mathrm{E}[k(z)d(y,f(z))^2]$ of the variational formulation has solutions which are denoted *weighted Fréchet means*, see e.g. [19] (local minimizers are denoted weighted Karcher means). The weighted Fréchet mean is in [5] used to define a manifold valued convolution operator: For $x_1,\dots,x_n \in M$ and weights $w_1,\dots,w_n \in \mathbb{R}$, the generalized convolution is $\operatorname{argmin}_{y\in M} \sum_{i=1}^N w_i d(y,x_i)^2$. This can be formulated in a continuous setting for $f : M \to M$, $x \in \mathcal{M}$ as $k * f(x) = \operatorname{argmin}_{y\in M} \mathrm{E}[k(x,z)d(y,f(z))^2)]$ where $k : M \times M \to \mathbb{R}$ is the kernel satisfying $\mathrm{E}[k(x,\cdot)] = 1$ for each x.[1]

Generally, computing the weighted Fréchet mean is expensive requiring solution of nested optimization problems: Each computation $d(y,x_i)^2$ for a candidate y includes an optimization to find the squared length of a minimizing geodesic on M, and this computation has to be iterated in each step of an iterative, gradient based optimization to find an optimal y. This is clearly not adequate for deep learning applications. [5] propose to use an inductive estimator that computes an estimate of the wFM by computation of $n-1$ geodesics between the candidate point and the input x_i. This computation is efficient in the cases where geodesics can be computed efficiently, e.g. in closed form. In this case, it is possible to backpropagate through the wFM estimation, and thereby to use the convolution layer in a standard deep network setup.

The Fréchet mean as used in geometric statistics has a cousin in the Brownian motion maximum likelihood mean (mlM, see e.g. [25, 29, 31]). This definition uses that the Euclidean expected value has an equivalent definition as the maximally likely center point of a normal distribution fitted to data: If $p_\theta : \mathbb{R}^d \to \mathbb{R}$ denotes the density of a normal distribution $\mathcal{N}(y,\sigma^2)$ with parameter $\theta = (y,\sigma^2)$, then the maximizer of the log-likelihood $\bar{\theta} = \operatorname{argmax}_\theta \mathrm{E}[\log p_\theta(X)]$ for an $\mathbb{R}^d$-valued random variable X has the expected value in the y-variable, i.e. $\bar{y} = \mathrm{E}[X]$. Since the normal distribution can be generalized to manifolds with the Riemannian Brownian motion (this is one among other generalizations, see [23]), an equivalent Riemannian definition of mean value is $\operatorname{argmax}_{y\in\mathcal{M}} \mathrm{E}[\log p_T(X;y)]$ where, as earlier on, $p_T(\cdot;y)$ denotes the solution of the heat flow started at $y \in M$, i.e. the density of the Riemannian Brownian motion, and T takes the role of the variance σ^2. The interest in the mlM lies in the natural incorporation of the curvature of M for data with large spread. In low dimensions, $p_T(\cdot;y)$ can be approximated directly using spectral methods, whereas in high dimensions, $p_T(\cdot;y)$ can be approximated by sampling the

[1] [5] defines the convolution for an M-valued random variable X by $\operatorname{argmin}_{y\in M} \mathrm{E}[k(X)d(y,X)^2)]$.

Brownian motion conditioned on hitting the data, see e.g. [30] and below.

3.2 Weighted Maximum Likelihood Mean

We here propose a scheme related to the use of the wFM for manifold-valued convolution, but we exchange the wFM with a *weighted* maximum likelihood mean (wmlM): In Euclidean space, the weighted mean equals the maximally likely center point y of independent samples $x_1, \dots, x_n$ from n normal distributions $\mathcal{N}(y, T/w_1)$, $\dots, \mathcal{N}(y, T/w_n)$. Again taking the Brownian motion as the manifold equivalent of the Euclidean normal distribution with density $p_{T/w_i}(\cdot; y)$, we here define the weighted maximum likelihood mean wmlM as $\operatorname{argmax}_y \sum_{i=1}^n \log p_{T/w_i}(x_i; y)$ (discrete version) and $\operatorname{argmax}_y \mathrm{E}[\log p_{T/k(z)}(z; y)]$ (continuous version). As we will see below, the probabilistic nature of the mean allows sampling from a distribution centered at the mean, thereby removing the need for optimization to find geodesics.

Similarly to the use of the wFM for convolution, the manifold-valued convolution using the wmlM is here

$$k * f(x) = \operatorname{argmax}_{y \in M} \mathrm{E}[\log p_{T/k(x,z)}(f(z); y)]$$

with $k : M \times M \to \mathbb{R}$ and $f : M \to M$.

3.3 Bridge Sampling for Likelihood Approximation

We now switch the roles of y and the data x_i: In the Euclidean setting, we consider the probability of observing y in each of the n distributions $\mathcal{N}(x_i, T/w_i)$ simultaneously. The distribution of y is then $\mathcal{N}(\frac{\sum_{i=1}^n w_i x_i}{\sum_{i=1}^n w_i}, \frac{T}{\sum_{i=1}^n w_i})$, i.e. again a normal distribution however centered at the weighted mean. On manifolds, it is computationally difficult to compute the wmlM directly similarly to the wFM, but it turns out we can sample from a manifold equivalent of the distribution of y. We achieve this by sampling a conditioned Brownian motion in the n-fold product manifold M^n. Below, we first discuss sampling the conditioned distribution in the one sample situation ($n = 1$, $w_1 = 1$) before moving on to the weighted case.

Let $y, v \in M$, and let X_t^y denote the Brownian motion starting at y. The time $t = T$ conditioned process $X_t^x | X_T = v$, a Brownian bridge, has the property of hitting the target value v at time T a.s. Sampling of the conditioned process is often of interest because it can be used to approximate the heat kernel $p_T(v; y)$. That is, if we can sample a process that approximates $X_t^y | X_T = v$, we can approximate the heat kernel even in high dimensions were direct solution of the heat PDE is not applicable. In coordinates, the conditioned process satisfy the Ito stochastic differential equation (SDE)

$$dx_t = b(x_t)dt + a(x_t)a(x_t)^T \nabla \log p_{T-t}(v; x_t)dt + a(x_t)dW(t) \tag{3.1}$$

where the drift $b(x) = -\frac{1}{2} g(x)^{kl} \Gamma(x)_{kl}$ involves a contraction over the Christoffel symbols, and the diffusion coefficient $a(x) = \sqrt{g^{-1}(x)}$ is a square root of the inverse of the Riemannian metric g. This SDE is however not useful for computational purposes since we cannot expect to be able to compute $\nabla \log p_{T-t}(v; x_t)$ at each

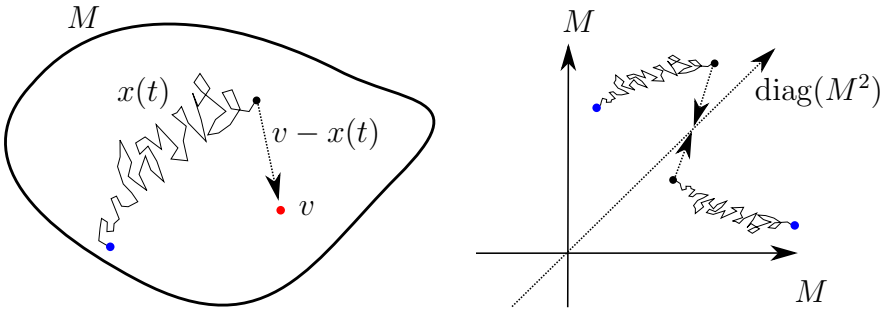

Figure 6: (left) On M, the guided proposal scheme (3.2) forces the process to hit the target v at time T a.s. by adding the drift $\tilde{b}$ that is typically the difference $v - x(t)$ (in coordinates, dotted arrow in figure) scaled by the inverse remaining time $(T-t)^{-1}$. The difference between the law of the conditioned and the guided process is proportional to the factor $\varphi(x_t)$. (right) On the product M^2 (or M^n), we can apply the guidance scheme to force independent Brownian motions to hit each other at time T. This is done by adding a drift $\tilde{b}$ that forces the processes towards the diagonal by adding the differences to the weighted arithmetic mean (in coordinates) of the processes. The result is the $\mathrm{diag}(M^2)$ valued random variable v.

time step. Instead, Delyon and Hu [9] introduced a guided approximation of the bridge process. This consist of an SDE

$$dx_t = b(x_t)dt + \tilde{b}(x_t)dt + a(x_t)dW(t) \tag{3.2}$$

where the term $a(x_t)a(x_t)^T \nabla \log p_{T-t}(v; x_t)$ above is exchanged with a computationally feasible alternative, either $\tilde{b}(x) = \frac{v-x}{T-t}dt$, or, alternatively,

$$\tilde{b}(x) = a(x_t)a(x_t)^T \nabla \log \tilde{p}_{T-t}(v; x)$$

where $\tilde{p}$ is a density of a simpler process with closed form transition density [26]. Here, we follow the former approach as used in [30] to sample the Brownian motion. With this, the transition density can be estimated by Monte Carlo sampling of the expectation $\mathrm{E}[\varphi(x_t)]$ because of the relation

$$p(T, v; y) = \sqrt{\frac{|(a(x)^{-1})^T a(x)|}{(2\pi T)^d}} e^{\frac{-\|a(y)^{-1}(y-v)\|^2}{2T}} \mathrm{E}[\varphi(x_t)]$$

where φ denotes a correction factor between the law of the true bridge and the guided proposal process. The guided proposal scheme is illustrated in Figure 6 (left).

3.4 Sampling the wmlM

However, the goal here is not to estimate $p_T(v; y)$ but to sample a distribution approximating the wmlM. We use the ideas above to turn the problem around in the following way: For observations $x_1, \ldots, x_n$, let M^n denote the product manifold with the product Riemannian metric. We then start a process $x_t = (x_{1,t}, \ldots, x_{n,t})$ at the point $(x_1, \ldots, x_n) \in M^n$ and condition it on having equal components at

time $t = T$, i.e. $x_{1,T}^n = \cdots = x_{n,T}^n$. That is, the process must hit the diagonal of the product space M^n similarly to the simultaneous observation of y with n normal distribution in the Euclidean situation described at the start of section 3.3. The processes $x_{i,t}$ are independent Brownian motions with variance T/w_i. This gives the conditioned process

$$(x_{1,t}, \ldots, x_{n,t}) | x_{1,T} = \cdots = x_{n,T} \ . \tag{3.3}$$

The conditioned process is analogous to the Brownian bridge except that we condition on a subspace in M^n instead of a point in M. In essence, the process runs backwards from the observations to reach a point $v = x_{1,T}^n = \cdots = x_{n,T}$ in M. Due to the symmetry $p_{T/w_i}(y; x) = p_{T/w_i}(x; y)$ of the Brownian motion, and the independence of the individual Brownian motions $x_{i,t}$ on M^n, we have $p_{T/w_i}(v; (x_1, \ldots, x_n)) = \prod_{i=1}^n p_{T/w_i}(v; x_i) = \prod_{i=1}^n p_{T/w_i}(x_i; v)$, i.e. the probability of observing v at the diagonal equals the probability of observing $x_1, \ldots, x_n$ on M regarding v as a parameter.

Similarly to (3.1), the conditioned process has an SDE that depends on the (intractable) log-transition density. However, we can again construct a guided process (3.2) on the product manifold M^n using a drift which in a coordinate chart reads $\tilde{b}(x_{1,t}, \ldots, x_{n,t}) = ((\mu(t) - x_{1,t})/w_1, \ldots, (\mu(t) - x_{n,t})/w_n)/(T - t)$ with $\mu(t) = \frac{\sum_{i=1}^n w_i x_i(t)}{\sum_{i=1}^n w_i}$. The scheme is illustrated in Figure 6 (right). We let φ denote the correction factor as above.

We then obtain the following result.

Theorem 3.1. *Let $x_t = (x_{1,t}, \ldots, x_{t,n})$ consist of n independent Brownian motions on M with variance T/w_i, and let $\tilde{x}_t$ be the process with additional added drift $\tilde{b}$. Let P be the law of $\tilde{x}_t$; P^* the law of the conditioned process* (3.3)*; and φ the correction factor of the guided process as in* (3.2)*. Let $v(x_1, w_1, \ldots, x_n, w_n)$ be the random variable $\tilde{x}_{1,T}$ with law $\frac{\varphi}{\mathrm{E}[\varphi]} P$. Then v has a density $p_v(y) = \prod_{i=1}^n p_{T/w_i}(x_i; y)$ and $v = \tilde{x}_{i,T}$ for all i a.s.*

Proof. That the construction of [9] extends to conditioning on subspaces is shown in [33]. The distribution of v with respect to the probability measure $\frac{\varphi}{\mathrm{E}[\varphi]} P$ equals the distribution of $x_{1,T}^n$ with respect to P^*. Because the processes $x_{i,t}$ are independent, $p_v(y) = \prod_{i=1}^n p_{T/w_i}(x_i; y)$. The differences $\mu(T) - x_{i,T}$ are 0 similarly to the case of [9, 33] showing that $x_{i,T} = x_{j,T}$ for all $i, j = 1, \ldots, n$. □

The theorem states that the Euclidean situation with normally distributed y persists in the manifold situation: The random variable v arise from observing the same value v in n independent Brownian motions. The weighting appears as a scaling of the variances of the individual processes. The proof as shown here omits stochastic analysis details including when the process shifts between coordinate charts. A rigorous proof will be presented in a future paper.

We can now sample an approximation of the the wmlM by accounting for φ with the following sampling importance resampler (SIR, Algorithm 1). The algorithm as listed is written in coordinates assuming relevant charts. Alternatively, if M is embedded as a subset of $\mathbb{R}^k$, $\mathbb{R}^k$ coordinates can be used. The algorithm requires the computation of the Christoffel symbols in the integration of x_t as is required for the numerical integration of geodesics. However, importantly, it removes any need for nested iterative optimization as is used for the wFM in cases where geodesics do not

Algorithm 1: wmlM estimation by SIR

sample J paths from the guided process $\tilde{x}_t^j$
compute correction factors $\varphi^1, \ldots, \varphi^J$
sample j from $1, \ldots, J$ with probability $\{\varphi^j / \sum_{j=1}^J \varphi^j\}$
return $v = \tilde{x}_{j,T}$

have closed form solutions. Changes to the weights w_i affect the sample paths x_t^j and corrections φ^j. The coupling between w_i and φ is however only through interaction between the guidance term and the Christoffel symbols. In practice, this can be ignored allowing backpropagation through the algorithm.

Note that because the wmlM approaches the wFM when T is small, samples from Algorithm 1 will in this case approach the wFM. However, small T will affect the probability of the samples (φ will tend to zero) because any deviance of the guided process from a geodesic will be less likely. Nonzero T can thus be seen as a way to get computational efficiency at the cost of variance in the estimator. Conversely, stochasticity can be reduced by lowering the evaluation time T of the Brownian motion. This will reduce the variance of v, and result in the wmlM approaching the wFM. However, this will require larger J in algorithm 1 and thus increase in computational cost.

3.5 Stochastic NN Outputs

While the stochasticity of the wmlM estimator adds randomness to an otherwise deterministic setup, the added stochasticity is rather natural. For example, the deep Gaussian process model employed in [13] when using dropout for uncertainty quantification uses the data conditional distribution $y|x, w = \mathcal{N}(\hat{y}(x, w), \tau^{-1})$ for the output y given the input x, weights w, deterministic neural network output $\hat{y}(x, w)$, and precision parameter τ. Here, we get the same distribution of v with $\tau^{-1} = T / \sum_{i=1}^n w_i$. $\sum_{i=1}^n w_i / T$ can therefore be regarded a precision parameter for the model. Comparing to Euclidean networks, because the stochasticity is built into the convolution operator, stochasticity is here added before application of a nonlinearity, while the output in [13] is normally distributed after application of nonlinearities in $\hat{y}(x, w)$. Similarly to the Monte Carlo sampling of the moments of the predictive distribution in [13], Algorithm 1 can be used to estimate the moments of the output by using the correction factors φ^j as importance sampling weights.

3.6 Conclusion

We can exchange the wFM as used for defining manifold valued convolutional layers with the wmlM estimator $v(x_1, w_1, \ldots, x_n, w_n)$. By making the output stochastic, the need for iterative optimization to find the wFM is removed, and the computational effort is therefore lower. This is in particular important for manifolds without closed form solutions for geodesics.

4 Final Conclusion and Outlook

The paper concerned the application of fiber bundle geometry and methods from stochastic analysis on manifolds in geometric deep learning in two cases: convolution with manifold domain, and convolution with manifold target. We showed how horizontal flows in the frame bundle provides a direct way of quantifying the role of curvature in the non-commutativity of the convolution when using parallel transport along minimizing geodesics. We then used this insight, the stochastic flows in the Eells-Elworthy-Mallivin construction of Brownian motion, and stochastic development, to define a new convolution operator that in a natural way constructs a distribution of orientations globally on the manifold. The anti-development of the Brownian motion allows kernels on $\mathbb{R}^d$ to be applied in a seamless way. In addition, the distribution of orientations in the frame bundle allows evaluation of fully connected layers that incorporates global information over the manifold without pooling over orientations.

In the second part of the paper, we showed how the weighted Brownian motion maximum likelihood mean can be used to define a convolution that takes values in a manifold. By conditioning a stochastic process in the n-fold product space M^n, we obtain a method for sampling from a distribution that centers at the wmlM. This removes the need for nested iterative optimization for computing the weighted Fréchet in cases where geodesics do not have closed form solution, and thereby allows the convolution operator to be applied on a much more general class of manifolds.

While we briefly discuss computational aspects and provide an algorithm for sampling the wmlM, this paper focuses on introducing the theoretical constructions and foundations for applying nonlinear stochastic methods in geometric deep learning. We hope that this will inspire further developments in the field, both in its theoretical foundation and in development of efficient algorithms.

Acknowledgments

The work presented in this article is supported by the CSGB Centre for Stochastic Geometry and Advanced Bioimaging funded by a grant from the Villum foundation, and the Novo Nordisk Foundation grant NNF18OC0052000.

References

[1] James Atwood and Don Towsley. Diffusion-convolutional Neural Networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pages 2001–2009, USA, 2016. Curran Associates Inc. ISBN 978-1-5108-3881-9.

[2] Davide Boscaini, Jonathan Masci, Emanuele Rodoià, and Michael Bronstein. Learning Shape Correspondence with Anisotropic Convolutional Neural Networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16, pages 3197–3205, USA, 2016. Curran Associates Inc. ISBN 978-1-5108-3881-9.

[3] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst. Geometric Deep Learning: Going beyond Euclidean data. *IEEE Signal Processing Magazine*, 34 (4):18–42, July 2017. ISSN 1053-5888. doi: 10.1109/MSP.2017.2693418.

[4] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral Networks and Locally Connected Networks on Graphs. *arXiv:1312.6203 [cs]*, December 2013.

[5] Rudrasis Chakraborty, Jose Bouza, Jonathan Manton, and Baba C. Vemuri. A Deep Neural Network for Manifold-Valued Data with Applications to Neuroimaging. In Albert C. S. Chung, James C. Gee, Paul A. Yushkevich, and Siqi Bao, editors, *Information Processing in Medical Imaging*, Lecture Notes in Computer Science, pages 112–124. Springer International Publishing, 2019. ISBN 978-3-030-20351-1.

[6] Miranda C. N. Cheng, Vassilis Anagiannis, Maurice Weiler, Pim de Haan, Taco S. Cohen, and Max Welling. Covariance in Physics and Convolutional Neural Networks. *arXiv:1906.02481 [hep-th, stat]*, June 2019.

[7] Taco Cohen and Max Welling. Group Equivariant Convolutional Networks. In *International Conference on Machine Learning*, pages 2990–2999, June 2016.

[8] Taco S. Cohen, Maurice Weiler, Berkay Kicanaoglu, and Max Welling. Gauge Equivariant Convolutional Networks and the Icosahedral CNN. *Proceedings of the International Conference on Machine Learning (ICML), arXiv:1902.04615*, 2019.

[9] Bernard Delyon and Ying Hu. Simulation of conditioned diffusion and application to parameter estimation. *Stochastic Processes and their Applications*, 116(11):1660–1675, November 2006. ISSN 0304-4149. doi: 10.1016/j.spa.2006.04.004.

[10] David Elworthy. Geometric aspects of diffusions on manifolds. In Paul-Louis Hennequin, editor, *École d'Été de Probabilités de Saint-Flour XV–XVII, 1985–87*, number 1362 in Lecture Notes in Mathematics, pages 277–425. Springer Berlin Heidelberg, 1988. ISBN 978-3-540-50549-5 978-3-540-46042-8.

[11] Michel Emery. *Stochastic Calculus in Manifolds*. Universitext. Springer Berlin Heidelberg, Berlin, Heidelberg, 1989. ISBN 978-3-540-51664-4 978-3-642-75051-9.

[12] M. Frechet. Les éléments aléatoires de nature quelconque dans un espace distancie. *Ann. Inst. H. Poincaré*, 10:215–310, 1948.

[13] Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. In *International Conference on Machine Learning*, pages 1050–1059, June 2016.

[14] Alvina Goh, Christophe Lenglet, Paul M. Thompson, and René Vidal. A nonparametric Riemannian framework for processing high angular resolution diffusion images and its applications to ODF-based morphometry. *NeuroImage*, 56(3):1181–1201, June 2011. ISSN 1053-8119. doi: 10.1016/j.neuroimage.2011.01.053.

[15] Elton P. Hsu. *Stochastic Analysis on Manifolds*. American Mathematical Soc., 2002. ISBN 978-0-8218-0802-3.

[16] Ivan Kolář, Jan Slovák, and Peter W. Michor. *Natural Operations in Differential Geometry*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1993. ISBN 978-3-642-08149-1 978-3-662-02950-3.

[17] Risi Kondor and Shubhendu Trivedi. On the Generalization of Equivariance and Convolution in Neural Networks to the Action of Compact Groups. *arXiv:1802.03690 [cs, stat]*, February 2018.

[18] Line Kühnel, Alexis Arnaudon, and Stefan Sommer. Differential geometry and stochastic dynamics with deep learning numerics. *arXiv:1712.08364 [cs, stat]*, December 2017.

[19] Yongdo Lim and Miklós Pálfia. Weighted inductive means. *Linear Algebra and its Applications*, 453:59–83, July 2014. ISSN 0024-3795. doi: 10.1016/j.laa.2014.04.002.

[20] J. Masci, D. Boscaini, M. M. Bronstein, and P. Vandergheynst. Geodesic Convolutional Neural Networks on Riemannian Manifolds. In *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, pages 832–840, December 2015. doi: 10.1109/ICCVW.2015.112.

[21] Richard Montgomery. *A Tour of Subriemannian Geometries, Their Geodesics and Applications*. American Mathematical Soc., August 2006. ISBN 978-0-8218-4165-5.

[22] Federico Monti, Davide Boscaini, Jonathan Masci, Emanuele Rodolà, Jan Svoboda, and Michael M. Bronstein. Geometric deep learning on graphs and manifolds using mixture model CNNs. *arXiv:1611.08402 [cs]*, November 2016.

[23] Xavier Pennec. Intrinsic Statistics on Riemannian Manifolds: Basic Tools for Geometric Measurements. *J. Math. Imaging Vis.*, 25(1):127–154, 2006.

[24] Adrien Poulenard and Maks Ovsjanikov. Multi-directional Geodesic Neural Networks via Equivariant Convolution. *ACM Trans. Graph.*, 37(6):236:1–236:14, December 2018. ISSN 0730-0301. doi: 10.1145/3272127.3275102.

[25] S. Said and J. H. Manton. Extrinsic Mean of Brownian Distributions on Compact Lie Groups. *IEEE Transactions on Information Theory*, 58(6):3521–3535, June 2012. ISSN 0018-9448. doi: 10.1109/TIT.2012.2185680.

[26] Moritz Schauer, Frank van der Meulen, and Harry van Zanten. Guided proposals for simulating multi-dimensional diffusion bridges. *Bernoulli*, 23(4A):2917–2950, November 2017. ISSN 1350-7265. doi: 10.3150/16-BEJ833.

[27] Stefan C. Schonsheck, Bin Dong, and Rongjie Lai. Parallel Transport Convolution: A New Tool for Convolutional Neural Networks on Manifolds. *arXiv:1805.07857 [cs, math, stat]*, May 2018.

[28] Nicholas Sharp, Yousuf Soliman, and Keenan Crane. The Vector Heat Method. *ACM Trans. Graph.*, 38(3), 2019.

[29] Stefan Sommer. Anisotropic Distributions on Manifolds: Template Estimation and Most Probable Paths. In *Information Processing in Medical Imaging*, volume 9123 of *Lecture Notes in Computer Science*, pages 193–204. Springer, 2015.

[30] Stefan Sommer and Sarang Joshi. Brownian Bridge Simulation and Metric Estimation on Lie Groups and Homogeneous Spaces. *in preparation*, 2018.

[31] Stefan Sommer and Anne Marie Svane. Modelling anisotropic covariance using stochastic development and sub-Riemannian frame bundle geometry. *Journal of Geometric Mechanics*, 9(3):391–410, June 2017. ISSN 1941-4889. doi: 10.3934/jgm.2017015.

[32] Stefan Sommer, Francois Lauze, Søren Hauberg, and Mads Nielsen. Manifold Valued Statistics, Exact Principal Geodesic Analysis and the Effect of Linear Approximations. In *ECCV 2010*, volume 6316. Springer, 2010.

[33] James Thompson. Brownian bridges to submanifolds. *arXiv:1604.05182 [math]*, April 2016.