

Sparse Randomized Approximation of Normal Cycles

Allen Paul*, Neill Campbell†, Tony Shardlow‡

March 4, 2025

Abstract

We develop a compression algorithm for the Normal-Cycles representations of shape, using the Nystrom approximation in Reproducing Kernel Hilbert Spaces and Ridge Leverage Score sampling. Our method has theoretical guarantees on the rate of convergence of the compression error, and the obtained approximations are shown to be useful for down-line tasks such as nonlinear shape registration in the Large Deformation Metric Mapping (LDDMM) framework, even for very high compression ratios. The performance of our algorithm is demonstrated on large-scale shape data from modern geometry processing datasets, and is shown to be fast and scalable with rapid error decay.

1 Introduction

In many real-world geometric learning applications, one wishes to build statistical models of shape variation. For example, in computational anatomy where shapes are most commonly available as discrete curves and surfaces of anatomical structures [14]. In order to fit such models, one requires a fidelity metric on shapes in order compare the data to shapes generated by the model, and tune the model parameters. In the vast majority of such applications shapes of interest are available in varying resolutions, and do not have a consistent parametrization, meaning that simple fidelity metrics relying on known correspondences cannot be used. As such, a large literature has been developed on point cloud based metrics such as Chamfer distance, as well as methods for extracting dense correspondences given two shapes. However, in general, point cloud distances are *not* geometric and can be very sensitive to noise in the data. Furthermore, dense correspondences may not be well defined or unique, which introduces a bias in subsequent shape learning applications based on choice of correspondence mapping used.

In the case where shape data are available as sub-manifolds of \mathbb{R}^d , one can deal with the lack of parametric correspondences in a more principled manner, using techniques from geometric measure theory [2]. In particular, using the so-called currents [13] and varifolds [3] representation of shapes. These representations view shapes as objects that integrate differential forms on the underlying domain. In dimensions $d = 2, 3$ by choosing these differential forms to lie in a Hilbert space, these representations essentially embed the shapes into a dual space of differential forms. Using the dual metric on these representations allows one to compute the distance between shapes, in terms of their action of vector fields and not in terms of their parametrizations. The computation of the dual metric between submanifolds can be written down explicitly when the Hilbert space is a Reproducing Kernel Hilbert space, induced by a choice of positive definite kernel function $k : X \times X \rightarrow \mathbb{R}$. One can also perform the same technique for discrete shape data in a way that is consistent as the resolution of the data tends to infinity. This framework has been used extensively in the LDDMM literature [2], for matching shapes with diffeomorphisms, using the functional shape metrics as the discrepancy term for matching.

While currents and varifolds allow for *geometric* comparison of shapes in a correspondence-less manner, these representations only take into account first order information about the geometric

*University of Bath ap2746@bath.ac.uk

†University of Bath n.campbell@bath.ac.uk

‡University of Bath T.Shardlow@bath.ac.uk

object they represent, such as tangential and normal direction information. If the structure we wish to describe have regions of high curvature, branching points or distinguished boundaries which are typically present in real-world applications, currents and varifolds metrics may not suffice in distinguishing such structures, as they are not sensitive to these properties of a shape. Indeed, it is observed in [2] that undesirable features may be generated during diffeomorphic registration, when these metrics are used as discrepancy terms for matching complex shapes with the aforementioned features.

A theoretically grounded solution to these issues is the concept of normal-cycle, which also originated [6] in the study of Geometric measure theory. The application of this representation for computational anatomy and LDDMM, was recently suggested in [10, 11]. The normal-cycle representation of shape is a generalization of the current associated to the shape. The current associated to a surface is a linear functional integrating test differential forms against the surface. On the other hand, a normal-cycle is defined as a current associated to the *unit normal bundle* of the surface. Therefore, the normal-cycle associated to a surface is a linear functional integrating test differential forms, over the unit normal bundle of the surface. As such, the normal-cycle representation has access to an enriched representation of the associated shape. Intuitively, the normal-cycle representation is not only sensitive to the rate of change of the surface, but also the rate of change of the normal vectors, which is of course connected to the curvatures of the underlying surface. This link is formalized in [10]. Indeed, one may even prove the Normal cycles representation ‘contains information’ about the mean and Gaussian curvatures of the surface locally.

In a similiar manner to currents and varifolds, one may compute the distance between two normal cycles by embedding into the dual of a suitable RKHS, and computing the dual metric which has closed form expressions. The advantages of comparing shapes with normal-cycles instead of currents/varifolds is evidenced extensively in numerical examples of [11], on shapes with high curvature regions, boundaries and branching points. However, much like the currents and varifolds metric between discrete shapes, the practical computation of the discrete normal-cycle metric is costly and scales as $\mathcal{O}(MN)$, when comparing triangulated surfaces with M, N triangles respectively, with much worse run-time in practice than currents/varifolds. In fact, it is observed heuristically in [11], that the normal-cycle representation is 8 – 10 times more expensive to compute and take gradients of than currents/varifolds and can become slow/memory consuming for large-scale problems. In modern shape modelling applications when M and N can exceed 10^5 , this is extremely intensive in both memory and computation. A recent development to allow large-scale exact metric computation (and therefore exact gradients), is using efficient GPU tiling schemes with CUDA and C++ as in the KeOps library [1]. This method allow scalable and fast metric computation up to a limit, typically up to 10^5 landmarks for normal-cycles. Past this size, the KeOps based method can become very slow, and memory intensive on the GPU, especially for shape matching and group-wise registration due to repeated metric and gradient computations. Indeed, this is already evidenced in the run-time comparison graphs of [11].

1.1 Contribution

In this work, we extend the compression algorithm of [9] for currents and varifolds to the normal-cycles setting, using the Nystrom approximation and interpolation theory in Reproducing Kernel Hilbert Spaces. In particular, we build on the work of [11], to derive an explicit Dirac delta basis form for discrete normal cycles, which allows us to apply the main theorem of [9]. This will allow us to compress target normal-cycles of effective size N , by forming sparse approximations of effective size n such that $n \ll N$. This is the first such algorithm for compression of normal-cycles, allowing one to use this theoretically attractive shape metric for the first time on large-scale problems. The theoretical guarantees on the convergence of the approximation derived in [9] automatically carry over to the normal-cycles setting. These guarantees are exponential error decay bounds of the form

$$\|\mu - \hat{\mu}\|_{W^*}^2 \leq \mathcal{O}(m \exp(-\alpha m^{\frac{1}{d}})),$$

for some $\alpha > 0$ and where $\hat{\mu}$ is the approximation of ‘size’ m to the original normal-cycle $\mu \in W^*$. Post compression, one can compute exact distances and gradients of distances between compressed

normal-cycles in $\mathcal{O}(mn)$, which gives significant savings when $m \ll M$, $n \ll N$. This can help massively speed up subsequent shape processing algorithms, as well reduce memory issues with such computations, even with KeOps. The benefits of our compression algorithm are demonstrated on LDDMM matching problems, with real world shapes that are massively over-sampled relative to the shape variation.

1.2 Existing Work

To the best of our knowledge, this is the first work to investigate *compression* of the Normal-Cycles representation, and scaling up metric computation to extremely dense large scale shape data. This work builds on the compression algorithm of [9] for Currents and Varifolds, and can be seen as an extension of the Ridge Leverage approximation technique applied there, to the more complex setting of the Normal Cycle. Indeed, much of the work we do in section 3, is to re-cast the normal-cycle representation in a Dirac Delta decomposition which allows one to apply the fast measure compression algorithm of [9]. While there has been previous work [5, 7] (other than [9]) on compressing geometric measures such as currents and varifolds, the compression techniques there are significantly slower than [9], and do not always come with theoretical guarantees. Hence, we choose to extend the work of [9] to the normal-cycle setting, resulting in a normal-cycle compression algorithm which is fast, and has theoretical guarantees on the compression error decay.

1.3 Outline

In section 2, we review the currents and normal-cycles representation of shape within the framework of exterior algebra, differential forms and rectifiable sets. In section 3, we build upon the work of [11] on normal cycles to derive an explicit Dirac delta form of the normal-cycle of a triangulated surface, so that one is in a position to apply the randomized compression algorithm of [9]. In section 3, we discuss the how one may compress the aforementioned Dirac delta functional form of normal-cycles and application to nonlinear registration in the LDDMM setting. Finally, in section 4 we demonstrate the strengths and weaknesses of the proposed compression algorithm on large-scale shape data from modern geometry processing datasets.

2 Background

We now define rigorously, the normal-cycle associated to a submanifold using the language of geometric measure theory. Throughout, we loosely follow the notation of [10], the original work which introduced the normal-cycle to computational anatomy and LDDMM. We begin by discussing the currents representation of shape, from which the normal-cycle can be obtained as a higher order extension.

2.1 Currents

The currents representation of shape originates in geometric measure theory [6], developed for the study of the Plateau problem. The core idea, is to represent shapes as objects that integrate continuous differential forms. Formally, one begins by defining the space of continuous compactly supported differential m -forms

$$\Omega_0^m(\mathbb{R}^d) = \mathcal{C}_0(\mathbb{R}^d, (\Lambda^m \mathbb{R}^d)^*),$$

equipped with the uniform norm. Subsequently, one defines the space of m -currents as $\Omega_0^m(\mathbb{R}^d)^*$, which is the topological dual of $\Omega_0^m(\mathbb{R}^d)$. This means that m -currents are bounded linear functionals $T : \Omega_0^m(\mathbb{R}^d) \rightarrow \mathbb{R}$ such that for all $\omega \in \Omega_0^m(\mathbb{R}^d)$, the following holds for some $C(T) > 0$:

$$|T(\omega)| \leq C(T) \|\omega\|_\infty.$$

This is summarized in the following definition

Definition 2.1. Given an m -Hausdorff rectifiable set $S \subset \mathbb{R}^d$, the unique current associated with it is defined as $[S] \in \Omega_0^m(\mathbb{R}^d)^*$ with action

$$[S](\omega) := \int_S (\omega(x) | \tau^S(x)) d\mathcal{H}^m(x), \quad \omega \in \Omega_0^m(\mathbb{R}^d), \quad (1)$$

where for \mathcal{H}^m almost every x in S , one defines $\tau^S(x) := e_1(x) \wedge \dots \wedge e_m(x)$, a wedge product of orthonormal basis vectors of the tangent space at x . This is well defined, as the tangent space exists \mathcal{H}^m a.e. for a m -Hausdorff rectifiable set, and the choice of basis vectors does not matter due to invariance of wedge product under change of orthonormal basis.

From eq. (1), we observe the current associated to S takes differential m -forms as input, and returns the integral of the differential form against the rectifiable set, with respect to the Hausdorff measure. When S is a smooth m -dimensional submanifold, this reduces to the standard definition of integration of m -forms on a smooth manifold [10]. The triangle inequality yields

$$|[S](\omega)| \leq \|\omega\|_\infty \mathcal{H}^m(S), \quad \omega \in \Omega_0^m(\mathbb{R}^d)$$

so that $[S]$ is indeed a well-defined element of $\Omega_0^m(\mathbb{R}^d)^*$. One can also define a natural push-forward action [10] $\varphi_\# [S] \in \Omega_0^m(\mathbb{R}^d)^*$ on currents under the action of a diffeomorphism $\varphi : \mathbb{R}^d \rightarrow \mathbb{R}^d$ on the ambient space. While we do not pursue this definition here, one may prove [10] that this action is geometric, in the sense that

$$\varphi_\# [S] = [\varphi(S)],$$

This identity makes the diffeomorphic pushforward of a current a straightforward object to handle. This is extremely useful for applications of currents in the LDDMM framework, where shapes are modelled as deformations of template shapes under the action of a diffeomorphism.

2.1.1 Dual space metrics for Currents

It was originally suggested in [13] that one may compare shapes in a non-parametric, correspondence-less manner, by comparing them in the the dual space metric on currents. The associated ‘distance’ between shapes is given as

$$d([S], [T]) := \|[S] - [T]\|_{\Omega_0^m(\mathbb{R}^d)^*} = \sup_{\omega \in \Omega} |[S](\omega) - [T](\omega)|. \quad (2)$$

In eq. (2), shapes are compared in terms of how similarly or differently they integrate the same differential forms, and one computes the maximum such discrepancy over all $\omega \in \Omega_0^m(\mathbb{R}^d)$. Unfortunately, the uniform norm topology over $\Omega_0^m(\mathbb{R}^d)$ is too strong and does not give a useful measure of dissimilarity to compare and register shapes as proven in [3]. Indeed, the distance eq. (2) simply collapses to the sum of Hausdorff measure of each shape.

Instead, as proposed by [13], one embeds currents of interest $[S], [T]$ into the dual of a smaller space $W \subset \Omega_0^m(\mathbb{R}^d)$ of differential forms, equipped with an RKHS structure. This induces a weaker topology on currents, and ensures the dual metric in W^* yields closed form expressions which one use for comparison/optimization purposes. In particular, [13] proposes to define W as an RKHS of differential forms, generated by a kernel $K : \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathcal{L}(\Lambda^m \mathbb{R}^d)$ of the form

$$K(x, y) = K_p(x, y) \text{Id}_{\Lambda^m \mathbb{R}^d},$$

where K_p is a positive-definite scalar-valued spatial kernel on \mathbb{R}^d . Such a W takes the form

$$W = \left\{ f = \sum_{i=1}^{\infty} K_p(\cdot, x_i) \tau_i : \tau_i \in \Lambda^m \mathbb{R}^d, \quad x_i \in \mathbb{R}^d, \quad \|f\|_W^2 = \sum_{i,j=1}^{\infty} K_p(x_i, x_j) \langle \tau_i, \tau_j \rangle_{\Lambda^m \mathbb{R}^d} < +\infty \right\}.$$

Provided that W is dense in $\Omega_0^m(\mathbb{R}^d)$ this guarantees the dual embedding $\Omega_0^m(\mathbb{R}^d)^* \subset W^*$ is an injection [10]. A popular choice of kernel function that ensures this is the Gaussian RBF kernel.

The action of $[S], [T]$ as elements of W^* is the same integral action as in eq. (1), and they are well-defined elements of W^* due to the bound

$$|[S](\omega)| \leq \|\omega\|_\infty \mathcal{H}^m(S) \leq c \|\omega\|_W \mathcal{H}^m(S), \quad \omega \in W,$$

where we have used the RKHS identity $\|\omega\|_\infty \leq c \|\omega\|_W$ for some $c > 0$. Using the reproducing property of the RKHS kernel, one may compute the dual metric in W^* between $[S], [T]$ explicitly as

$$d_{W^*}([S], [T])^2 = \|[S] - [T]\|_{W^*}^2 = \langle [S], [S] \rangle_{W^*} - 2\langle [S], [T] \rangle_{W^*} + \langle [T], [T] \rangle_{W^*}, \quad (3)$$

where the product terms are defined as

$$\langle [S], [T] \rangle_{W^*} = \int_S \int_T K_p(x, y) \langle \tau^S(x), \tau^T(y) \rangle_{\Lambda^m \mathbb{R}^d} d\mathcal{H}^m(x) d\mathcal{H}^m(y). \quad (4)$$

In the discrete case, when we have polygonal approximations \hat{S}, \hat{T} of underlying shapes S, T , one forms approximate currents

$$[\hat{S}] \approx [\widehat{S}] := \sum_{i=1}^n \delta_{c_{i,S}} \tau_{i,S} \in W^*, \quad [\hat{T}] \approx [\widehat{T}] := \sum_{i=1}^m \delta_{c_{i,T}} \tau_{i,T} \in W^*,$$

where $c_{i,S}, c_{i,T}$ denote the centre of the i 'th polygon of \hat{S}, \hat{T} respectively. The weights $\tau_{i,S}, \tau_{i,T}$ are defined as $\tau_{\hat{S}}(c_{i,S}), \tau_{\hat{T}}(c_{i,T})$ respectively. Applying the reproducing property of the RKHS kernel yields a simple to compute dual metric with 'discrete' product term

$$\langle [\widehat{S}], [\widehat{T}] \rangle_{W^*} = \sum_{i=1}^n \sum_{j=1}^m K_p(c_{i,S}, c_{j,T}) \langle \tau_{i,S}, \tau_{j,T} \rangle_{\Lambda^m \mathbb{R}^d}. \quad (5)$$

This may be interpreted as the discretisation of eq. (4). Such delta approximations are known to be consistent [2] in the sense that

$$\|[\widehat{S}] - [S]\|_{W^*} \lesssim \tau(h)$$

where the bounding function satisfies $\tau(h) \rightarrow 0$, as h the discretization size of the approximation tends to zero. This yields the discrete currents metric

$$d_{W^*}([\widehat{S}], [\widehat{T}])^2 = \|[\widehat{S}] - [\widehat{T}]\|_{W^*}^2 = \langle [\widehat{S}], [\widehat{S}] \rangle_{W^*} - 2\langle [\widehat{S}], [\widehat{T}] \rangle_{W^*} + \langle [\widehat{T}], [\widehat{T}] \rangle_{W^*},$$

In order to practically compute the weights τ_S, τ_T in eq. (5), one observes first that $\Lambda^m \mathbb{R}^d \simeq \mathbb{R}^{\binom{d}{m}}$. For surfaces, we have $(d, m) = (3, 2)$ so that we may identify m -forms with vector fields, and the weights τ with three dimensional vectors. In fact, under this identification, for triangulated surfaces $\tau_{i,S}(x)$ correspond to the normal vector at $x \in S$ scaled by the area of the approximating triangle which is simple to compute. For discrete curves, we have $(d, m) = (3, 1)$ or $(d, m) = (2, 1)$ corresponding to three dimensional and two dimensional curves respectively. In either case, the weight $\tau_{i,S}$ reduce to the discrete tangent vector of the i 'th discrete curve segment. The discrete currents metric between shapes has been widely used [2] computational anatomy and LDDMM for comparison/registration of anatomical curves and surfaces without correspondence, as the above metric do not depend on particular parametrizations of S, T .

2.2 Normal Cycles

While the currents representation is appealing, there are limitations due to it's first-order nature; specifically, when using the currents metric as a discrepancy term for shape registration. Indeed, currents often struggle [2] in matching shapes with high curvature regions, and boundaries that are required to be enforced. This is where the Normal Cycle representation [11] is useful. The

normal-cycle associated with S , is the current associated to its unit normal bundle. Integration against the normal bundle allows one to account for the rate of change of the normals (curvature), as well as the rate of change of position. Indeed, normal-cycle representation is provably ‘sensitive’ [10] to curvature information, and yields a practically better [11] performing shape metric, when dealing with shapes with high curvature regions and boundaries.

In this section, we define the Normal-Cycle of a submanifold following [11]. We use the following notation of [11] in this section

$$\Omega_0^{d-1}(\mathbb{R}^d \times \mathbb{S}^{d-1}) = \mathcal{C}_0(\mathbb{R}^d \times \mathbb{S}^{d-1}, (\Lambda^{d-1}(\mathbb{R}^d \times \mathbb{R}^d))^*)$$

for $d - 1$ differential forms on $\mathbb{R}^d \times \mathbb{S}^{d-1}$.

2.2.1 Unit Normal Bundle and Normal Cycle of a smooth submanifold

For *smooth* submanifolds $S \subset \mathbb{R}^d$, the standard way to define unit normal bundle is as follows,

$$\mathcal{N}_S = \{(x, n) : x \in S, n \in (T_x S)^\perp, \|n\| = 1\}$$

where the tangent spaces are defined in the standard way, and the orthogonality is with respect to the ambient Euclidean metric. If S is an m -dimensional manifold, the unit normal bundle is a $d - 1$ -dimensional submanifold of $\mathbb{R}^d \times \mathbb{S}^{d-1}$ [10], independent of the dimension of m . For example, given a surface or space curve S in \mathbb{R}^3 , the normal bundle can be identified with a 2 dimensional manifold in $\mathbb{R}^3 \times \mathbb{S}^2$. Indeed, in either case one may visualise the normal as a smooth ‘fattening’ [11] of the underlying set. Since the unit normal bundle is a smooth $d - 1$ submanifold of $\mathbb{R}^d \times \mathbb{S}^{d-1}$, it is rectifiable and so the notion of current associated to the unit normal bundle is well defined; this is what is called the normal-cycle associated to S .

Definition 2.2. Given a smooth submanifold $S \subset \mathbb{R}^d$, the normal cycle associated to S is defined as $N(S) := [\mathcal{N}_S]$ which is an element of $\Omega_0^{d-1}(\mathbb{R}^d \times \mathbb{S}^{d-1})^*$. The action of $N(S)$ on $\omega \in \Omega_0^{d-1}(\mathbb{R}^d \times \mathbb{S}^{d-1})$ is given by

$$N(S)(\omega) = [\mathcal{N}_S](\omega) = \int_{\mathcal{N}_S} (\omega(x, \nu) | \tau^{\mathcal{N}_S}(x, \nu)) d\mathcal{H}^{d-1}(x, \nu). \quad (6)$$

where $\tau^{\mathcal{N}_S}(x, \nu) = e_1(x, \nu) \wedge \cdots \wedge e_{d-1}(x, \nu)$ is the wedge product of an orthonormal basis of the normal bundle at point (x, ν) .

Remarkably, it is possible to show [10] that there exists differential forms such that the evaluation of the normal cycle on these forms retrieve local curvature information of the underlying shape, thus justifying the reference to the normal cycle as a curvature measure associated to S .

2.2.2 Normal Cycle of a union of smooth sets

In practical geometric learning, we work with discrete shapes in the form of discrete curves and triangulated surfaces, not smooth submanifolds. As such, one needs to define the normal cycle for such discrete objects, similar to discrete currents. In full generality the normal-cycle may be defined for sets of ‘positive reach’ [11] which includes unions of smooth submanifolds as a subset. However, we do not pursue this construction here, and instead rely on the following definition of [10].

Definition 2.3. If $S = S_1 \cup S_2$ is a union of C^2 manifolds with boundary such that $S_1 \cap S_2$ is smooth, then one defines

$$N(S) := N(S_1) + N(S_2) - N(S_1 \cap S_2) \quad (7)$$

In order to compute the normal-cycle $N(\mathcal{T})$ of a triangulated surface $\mathcal{T} = \cup_{i=1}^{n_T} T_i$ given as a union of closed triangles, one applies the formula eq. (7) recursively to obtain a decomposition in terms of the normal cycles of individual triangles and segments. Indeed, [11] derives an explicit expression for $N(\mathcal{T})$ as a sum of normal-cycles associated to disjoint components of the original $N(\mathcal{T})$. We shall use this decomposition in section 3. The definition is also consistent [10] with the general definition of $N(\mathcal{T})$ for sets of positive reach, thus justifying the resulting decomposition.

2.2.3 Dual Space Metric - Normal Cycles

Similar to the case of Currents, the topology on the dual space induced by the uniform norm is too strong to give a meaningful distance between shapes for purposes of shape comparison/registration. As for currents, fixing an appropriate RKHS W of test differential forms, one embeds the Normal Cycle representation into the dual W^* . This allows one to tractably compute RKHS dual norms in terms of the kernel function.

One begins, by defining a $\mathcal{L}(\Lambda^{d-1}(\mathbb{R}^d \times \mathbb{R}^d))$ valued positive definite kernel on $\mathbb{R}^d \times \mathbb{S}^{d-1}$ as

$$K((x, u), (y, v)) = K_p(x, y)K_s(u, v)\text{Id}_{\Lambda^{d-1}(\mathbb{R}^d \times \mathbb{R}^d)}, \quad \forall (x, u), (y, v) \in \mathbb{R}^d \times \mathbb{S}^{d-1}.$$

This induces an RKHS $W \subset \Omega_0^{d-1}(\mathbb{R}^d \times \mathbb{S}^{d-1})$ of forms $\omega : \mathbb{R}^d \times \mathbb{S}^{d-1} \rightarrow \Lambda^{d-1}(\mathbb{R}^d \times \mathbb{R}^d)$. The space W takes the form,

$$W = \left\{ f = \sum_{i=1}^{\infty} K_p(\cdot, x_i)K_s(\cdot, s_i)\tau_i : \tau_i \in \Lambda^{d-1}(\mathbb{R}^d \times \mathbb{R}^d), \quad (x_i, s_i) \in \mathbb{R}^d \times \mathbb{S}^{d-1}, \right. \\ \left. \|f\|_W^2 = \sum_{i,j=1}^{\infty} K_p(x_i, x_j)K_s(s_i, s_j)\langle \tau_i, \tau_j \rangle_{\Lambda^{d-1}(\mathbb{R}^d \times \mathbb{R}^d)} < +\infty \right\}.$$

Typical choices for the spherical kernel are, constant $K_s(u, v) = 1$, linear $K_s(u, v) = \langle u, v \rangle$, and Gaussian $K_s(u, v) = \exp(-\frac{\|u-v\|^2}{2\sigma^2})$. It is proven in [10], that for surfaces, the kernel metric is sensitive to mean curvature when K_s is constant, and Gaussian curvature when K_s is linear. The spatial kernel K_p is typically chosen to be a Gaussian RBF kernel. With the same action as eq. (6) on elements of W , the normal cycle $N(S)$ may be embedded as a well-defined element of W^* .

By the reproducing property of the kernel, the dual metric may be expressed [11] as

$$\|N(S) - N(T)\|_{W^*}^2 = \langle N(S), N(S) \rangle_{W^*} - 2\langle N(S), N(T) \rangle_{W^*} + \langle N(T), N(T) \rangle_{W^*},$$

with product term

$$\langle N(S), N(T) \rangle_{W^*} := \int_{\mathcal{N}_S} \int_{\mathcal{N}_T} K_p(x, y)K_s(u, v)\langle \tau_S(x, u), \tau_T(y, v) \rangle d\mathcal{H}^{d-1}(x, u)d\mathcal{H}^{d-1}(y, v), \quad (8)$$

where $\tau_S(x, u), \tau_T(x, u)$ are defined as in eq. (6).

The authors of [11] approximate the inner product terms eq. (8) *consistently*, in order to derive approximations for the normal cycles metric. Explicit closed form approximations of kernel metrics over normal cycles, are computed by [11] for discrete surfaces in the case of $K_s = 1$. This is the content of the following theorem.

Theorem 1. [11] *Let $\mathcal{T}, \mathcal{T}'$ be two triangulated surfaces. The inner product between the associated discrete Normal Cycles can be computed as,*

$$\langle N(\mathcal{T}), N(\mathcal{T}') \rangle_{W^*} = \frac{\pi^2}{4} \sum_{i=1}^{n_e} \sum_{j=1}^{m_e} k_p(c_i, d_j) \langle f_i, g_j \rangle \left\langle \sum_{\{T|f_i \text{ edge of } T\}} n_{T,f_i}, \sum_{\{T|g_j \text{ edge of } T\}} n_{T,g_j} \right\rangle \quad (9) \\ + \frac{\pi^2}{4} \sum_{\substack{x_i \text{ vertex} \\ \text{of } \partial\mathcal{T}}} \sum_{\substack{y_j \text{ vertex} \\ \text{of } \partial\mathcal{T}'}} k_p(x_i, y_j) \langle A_i, B_j \rangle \quad (10)$$

where $\partial\mathcal{T}$ denotes the boundary of the surface \mathcal{T} and similarly for \mathcal{T}' . In the above, n_e, m_e denotes the number of unique edges in each triangulation, and f_i, g_j the unique edges respectively, assuming an ordering on the vertices. The kernel function in the first double sum is evaluated on c_i, d_j which denote centres of the edges f_i, g_j respectively. In the second double sum, the kernel function is evaluated on x_i, y_j which are vertices of the boundary. Furthermore, the quantity $A_i := \sum_k f_k^i / |f_k^i|$ is defined as the sum of normalized edges attached to the boundary vertex x_i , oriented outwards from x_i . Finally, n_{T,f_i} denotes the normal to triangle T such that $n_{T,f_i} \times f_i$ is oriented inwards for the triangle T .

Note that the normal-cycle metric has an explicit term for comparing boundaries of shapes, unlike currents and varifolds. Furthermore, the inner product of sum of normals and edges, are closely connected to the discrete mean curvature [10].

The computational complexity of computing eq. (9) is much higher than that of currents. The first term is a double loop over edges, of which there are more than the number of triangles, and each term involves more inner product computations than for currents. Furthermore, the added boundary term also contributes an additional cost. It is observed in [11], that the practical cost of computing the metric and its gradient is around 8 – 10 times higher for normal-cycles compared to currents and varifolds. This cost is dealt with in [11] using the PyKeops package developed for large-scale kernel learning applications. However, in modern geometry processing tasks where discretized shapes \hat{S}_1, \hat{S}_2 are available with resolutions $N, M \geq 10^5$, even KeOps based implementations can become extremely slow due to memory limitations, and computational complexity of the metric computation.

For curves, the situation is different and one may compute the approximation to the product in the case of constant and linear spherical kernels, while the Gaussian case may be further approximated. The work of [10], yields the following normal-cycle metric on discrete curves when $K_s = 1$.

Theorem 2 ([10]). *Let $C = \{x_i\}_{i=1}^N$, $S = \{y_j\}_{j=1}^M$ be two discrete curves, given as sets of N and M vertices and associated edges respectively. Fixing the spatial kernel as K_p and constant normal kernel $K_s = 1$, the inner product between the associated normal cycles $N(C), N(S)$ is given by*

$$\langle N(C), N(S) \rangle_{W^*} = \frac{\pi^2}{4} \sum_{i=1}^N \sum_{j=1}^M k_p(x_i, y_j) \langle A_i, B_j \rangle,$$

where $A_i = \sum_k f_k^i / |f_k^i|$ is the sum of normalized edges oriented outwards from vertex x_i , and oriented outwards. The discrete edges are simply computed as $f_k^i = x_k - x_i$ for all vertices x_k connected to x_i in the discrete curve.

3 Decomposition and compression of Normal-Cycles

In order to apply the compression algorithm of [9] to discrete normal-cycles, we are required to compute an explicit Dirac delta decomposition in the dual space. In this section, we review the decomposition of the Normal-cycle into distinct components derived in [12, 10], and describe how one may explicitly compute a Dirac delta embedding with real-valued weights for $K_s = 1$. Subsequently, we show how to use this for compression of large-scale normal-cycles.

3.1 Curves

For discrete curves, the Dirac delta decomposition of the associated normal-cycle is straightforward and has already been observed in [10]. Indeed, [11] notes that given discrete curves C, S and $K_s = 1$, one may represent

$$N(C) = \sum_{i=1}^n \delta_{x_i} A_i \in W^*, \quad N(S) = \sum_{i=1}^m \delta_{y_i} B_i \in W^*$$

where we use the notation of eq. (9) for x_i, y_i, A_i, B_i , and W is an RKHS of vector fields induced by spatial kernel K_p . Similar decomposition for discrete curve normal-cycles may be derived [11], when K_s is linear or Sobolev kernel.

3.2 Surfaces

We now proceed with the case of deriving a Delta decomposition for the normal-cycle of discrete triangulated surfaces. Given a triangulation $\mathcal{T} = \cup_{i=1}^n T_i \subset \mathbb{R}^3$, a decomposition of $N(\mathcal{T})$ into distinct ‘planar’, ‘cylindrical’ and ‘spherical’ components is derived in [11] such that

$$N(\mathcal{T}) = N(\mathcal{T})^{pln} + N(\mathcal{T})^{cyl} + N(\mathcal{T})^{sph}, \quad (11)$$

and the three components are mutually orthogonal in W^* . The derivation given in [10] is long, and proceeds by recursively applying identity eq. (7) to the union of triangles $\mathcal{T} = \cup_{i=1}^n T_i$ to obtain the decomposition. In the following sections, we will explicitly compute Dirac delta sum form expressions for each component of $N(\mathcal{T})$.

We assume henceforth, that W is induced by a product kernel K , composed of an arbitrary spatial kernel K_p and constant spherical kernel $K_s = 1$. This is the setting explored in [11], for deriving explicit expressions for the normal-cycles metric. With this choice, one observes that generic elements $w \in W$ are of the form

$$w(x, n) = \sum_{i=1}^{\infty} K_p(x_i, x) K_s(n_i, n) \tau_i = \sum_{i=1}^{\infty} K_p(x_i, x) \tau_i, \quad (x_i, n_i, \tau_i) \in \mathbb{R}^d \times \mathbb{S}^{d-1} \times \Lambda^2(\mathbb{R}^d \times \mathbb{R}^d)$$

so that elements of W are constant on \mathbb{S}^{d-1} . In the following, we also denote $\omega_{x,n} := \omega(x, n)$, and further write $\omega_{x,n} = \omega_x$ in cases where wish to emphasize the invariance over \mathbb{S}^{d-1} .

3.2.1 Planar component

In [11] $N(\mathcal{T})^{pln}$ is derived to be the following sum of currents on the normal bundle:

$$N(\mathcal{T})^{pln} = \sum_{i=1}^{n_T} [T_i \times [\pm n_{T_i}]].$$

In the above, n_T is the number of triangles of \mathcal{T} , T_i is the i 'th triangle and n_{T_i} the outward unit normal to the i 'th triangle face. The individual terms in this sum are of the form,

$$\mathcal{C} = [T \times \{\pm n_T\}]$$

for some triangle T . This component of the normal-cycle has the action

$$\mathcal{C}(\omega) = \int_T \omega_{x,+n_T}(\tau_{x,n_T}) d\mathcal{H}^2(x) + \int_T \omega_{x,-n_T}(\tau_{x,-n_T}) d\mathcal{H}^2(x), \quad \omega \in W.$$

Recall that $\tau_{x,u}$ is constructed as a wedge product of an orthonormal basis of the tangent space at (x, u) of the normal bundle. Indeed one may show [11] that

$$\tau_{x,u} = \begin{pmatrix} e_1(x, u) \\ 0 \end{pmatrix} \wedge \begin{pmatrix} e_2(x, u) \\ 0 \end{pmatrix}$$

where $(e_1(x, u), e_2(x, u), u)$ is a positively oriented orthonormal basis of \mathbb{R}^3 . Since the differential forms in W have constant normal component, one has that $\omega_{x,+n_T} = \omega_{x,-n_T}$ for all $x \in T$. In addition, using the identity $\tau_{x,-u} = -\tau_{x,u}$, the bilinearity and antisymmetry of $\omega_{x,n}$ as a function of it's argument, we have

$$\begin{aligned} \mathcal{C}(\omega) &= \int_T (\omega_{x,+n_T}(\tau_{x,n_T}) + \omega_{x,-n_T}(\tau_{x,-n_T})) d\mathcal{H}^2(x) = \int_T (\omega_{x,+n_T}(\tau_{x,n_T}) + \omega_{x,+n_T}(\tau_{x,-n_T})) d\mathcal{H}^2(x) \\ &= \int_T (\omega_{x,+n_T}(\tau_{x,n_T}) + \omega_{x,+n_T}(-\tau_{x,+n_T})) d\mathcal{H}^2(x) = \int_T (\omega_{x,n_T}(\tau_{x,n_T}) - \omega_{x,n_T}(\tau_{x,n_T})) d\mathcal{H}^2(x) = 0, \end{aligned}$$

for all $\omega \in W$. As a consequence,

$$\mathcal{C} = 0 \implies N(\mathcal{T})^{pln} = 0.$$

since $N(\mathcal{T})^{pln}$ is a sum of such terms. The fact that the planar component vanishes for $K_s = 1$ is observed in [12].

3.2.2 Cylindrical component

It is shown in [11], that the cylindrical component of $N(\mathcal{T})$ is given as the following sum of currents on the normal bundle,

$$N(\mathcal{T})^{cyl} = - \sum_{j=1}^{n_e} \sum_{T \text{ triangle of } e_j} [[x_i, x_{i+1}] \times S_{f_i, -f_i \times n_T}^{+, \perp}] + \sum_{j=1}^{n_e} [\tilde{e}_j \times ((x_{j+1} - x_j)^\perp \cap S^{d-1})]. \quad (12)$$

where n_e denotes the total number of edges in the triangulation, $e_j = [x_j, x_{j+1}]$ denotes an edge in the triangulation with vertices x_j, x_{j+1} , and $\tilde{e}_j = e_j - \{x_j, x_{j+1}\}$. In the notation of [11], one defines

$$v^\perp := \{w \in \mathbb{R}^d : v \cdot w = 0\}, \quad S_{a,b}^{+, \perp} := (S^{d-1} \cap a^\perp) \cap (\{u : u \cdot b \geq 0\}),$$

so that $S_{a,b}^{+, \perp}$ is a half circle in the plane perpendicular to a , and oriented in the positive b direction. One may visualise $[x_i, x_{i+1}] \times S_{f_i, -f_i \times n_T}^{+, \perp}$ as a half cylinder centred on the edge $[x_i, x_{i+1}]$ oriented inwards to the edge. One may also visualize $\tilde{e}_j \times ((x_{j+1} - x_j)^\perp \cap S^{d-1})$ as a cylinder centred around the edge e_j .

The term $N(\mathcal{T})^{cyl}$ is a sum of cylindrical currents of the form

$$\mathcal{C} = [e \times C]$$

for closed/open edges e of the triangulation, and oriented half/full circles C centred on edge e . The action of \mathcal{C} on $\omega \in W$ is given by

$$\mathcal{C}(\omega) = \int_{e \times C} \omega_{x,u}(\tau_{x,u}) d\mathcal{H}^2(x, u)$$

As for for currents and varifolds, in order to allow practical metric computation, [11] approximates cylindrical components of the normal-cycle, at the centre c of edge e such that,

$$\mathcal{C}(\omega) \approx \hat{\mathcal{C}}(\omega) := |e| \int_C \omega_{c,u}(\tau_{c,u}) d\mathcal{H}^1(u) = |e| \int_C \omega_c(\tau_{c,u}) d\mathcal{H}^1(u) = |e| \omega_c \left(\int_C \tau_{c,u} d\mathcal{H}^1(u) \right),$$

which is a consistent approximation [11], as the edge-lengths tend to zero. One may write the approximation as

$$\hat{\mathcal{C}} = \delta_c |e| \alpha \in W^*, \quad \alpha := \int_C \tau_{c,u} d\mathcal{H}^1(u) \in \Lambda^2(\mathbb{R}^d \times \mathbb{R}^d), \quad (13)$$

with action

$$\hat{\mathcal{C}}(\omega) = \omega_c(|e| \alpha) = |e| \omega_c(\alpha).$$

The term α in eq. (13) may also be interpreted as the element in $\Lambda^2(\mathbb{R}^d \times \mathbb{R}^d) \simeq \Lambda^2(\mathbb{R}^d \times \mathbb{R}^d)^*$ such that

$$\langle \alpha, \beta \rangle = \int_C \langle \beta, \tau_{c,u} \rangle_{\Lambda^2(\mathbb{R}^d \times \mathbb{R}^d)} d\mathcal{H}^1(u), \quad \forall \beta \in \Lambda^2(\mathbb{R}^d \times \mathbb{R}^d)$$

To compute the integral for α , one observes [11] that $\tau_{x,u}$ has the following form on cylindrical components,

$$\tau_{x,u} = \begin{pmatrix} e_1(x, u) \\ 0 \end{pmatrix} \wedge \begin{pmatrix} 0 \\ e_2(x, u) \end{pmatrix} \quad (14)$$

where $(e_1(x, u), e_2(x, u), u)$ is a positively oriented orthonormal basis of \mathbb{R}^3 . In particular, note that at a given point (x, u) on the cylindrical component, such a basis may be computed as

$$(e_1(x, u), e_2(x, u), u) = \left(\frac{f}{|f|}, \frac{f}{|f|} \times u, u \right).$$

We note that this expression is indeed independent of x the first component along the cylinder. Substituting this choice of basis into equations eqs. (13) and (14) and applying bilinearity of the wedge product yields,

$$\begin{aligned}\int_C \tau_{c,u} d\mathcal{H}^1(u) &= \int_C \begin{pmatrix} \frac{f}{|f|} \\ 0 \end{pmatrix} \wedge \begin{pmatrix} 0 \\ \frac{f}{|f|} \times u \end{pmatrix} d\mathcal{H}^1(u) = \begin{pmatrix} \frac{f}{|f|} \\ 0 \end{pmatrix} \wedge \int_C \begin{pmatrix} 0 \\ \frac{f}{|f|} \times u \end{pmatrix} d\mathcal{H}^1(u) \\ &= \begin{pmatrix} \frac{f}{|f|} \\ 0 \end{pmatrix} \wedge \left(\int_C \frac{f}{|f|} \times u d\mathcal{H}^1(u) \right) = \begin{pmatrix} \frac{f}{|f|} \\ 0 \end{pmatrix} \wedge \begin{pmatrix} 0 \\ \frac{f}{|f|} \times \int_C u d\mathcal{H}^1(u) \end{pmatrix}\end{aligned}$$

The unit normal integral in the last term, is computed in [10] as

$$\int_C u d\mathcal{H}^1(u) = \begin{cases} 0, & \text{full-circle} \\ \frac{\pi}{2} \beta_T, & \text{half-circle} \end{cases}$$

where β_T is defined as the unitary vector orthogonal to the edge e , lying in the same plane as T , and oriented inwards to T . This finally yields

$$\alpha = \int_C \tau_{c,u} d\mathcal{H}^1(u) = \begin{cases} 0, & \text{full-circle} \\ \frac{\pi}{2} \begin{pmatrix} \frac{f}{|f|} \\ 0 \end{pmatrix} \wedge \begin{pmatrix} 0 \\ \frac{f}{|f|} \times \beta_T \end{pmatrix}, & \text{half-circle} \end{cases}.$$

The above implies that the full cylinder terms of eq. (12) are zero, so the second summation in eq. (12) vanishes. This leaves us with the half cylinder terms in the first summation of eq. (12), which upon substitution yields

$$N(\mathcal{T})^{cyl} = -\frac{\pi}{2} \sum_{j=1}^{n_e} \delta_{c_j} \gamma_j$$

where the weights are defined as

$$\gamma_j := \sum_{T \text{ triangle of } e_j} \begin{pmatrix} \frac{f_j}{|f_j|} \\ 0 \end{pmatrix} \wedge \begin{pmatrix} 0 \\ f_j \times \beta_T \end{pmatrix}.$$

3.2.3 Spherical Component

Finally, $N(\mathcal{T})^{sph}$ is shown in [11] to be equivalent to the normal cycle of the (possibly empty) boundary of the triangulation:

$$N(\mathcal{T})^{sph} = N(\partial\mathcal{T})^{sph},$$

with explicit expression

$$N(\partial\mathcal{T})^{sph} := \sum_{x_k \in \partial\mathcal{T}} [\{x_k\} \times \mathbb{S}^2] + \sum_{x_k \in \partial\mathcal{T}} \sum_{f \text{ edge of } x_k} [\{x_k\} \times S_{f/|f|}^+]. \quad (15)$$

In eq. (15), one denotes

$$S_v^+ = \{u \in \mathbb{S}^2 : u \cdot v \geq 0\},$$

for oriented half-spheres. Therefore, a generic element of the spherical component takes the form

$$\mathcal{C} = [\{x\} \times S],$$

for a full or half sphere S . Such terms have the following action

$$\mathcal{C}(\omega) = \int_S \omega_{x,u}(\tau_{x,u}) d\mathcal{H}^2(u) = \omega_x \left(\int_S \tau_{x,u} d\mathcal{H}^2(u) \right).$$

which may be rewritten in Dirac delta form as,

$$\mathcal{C} = \delta_x \alpha, \quad \alpha := \int_S \tau_{x,u} d\mathcal{H}^2(u)$$

On the spherical component $\tau_{x,u}$ has the following form on spherical components

$$\tau_{x,u} = \begin{pmatrix} 0 \\ e_1(x,u) \end{pmatrix} \wedge \begin{pmatrix} 0 \\ e_2(x,u) \end{pmatrix},$$

where $(e_1(x,u), e_2(x,u), u)$ is a positively oriented orthonormal basis of \mathbb{R}^3 . We may compute an explicit expression for such components by expressing $(e_1(x,u), e_2(x,u), u)$ in the standard basis of \mathbb{R}^3 . Indeed for fixed x, u one may express,

$$e_1(x,u) = \sum_{i=1}^3 a_i e_i, \quad e_2(x,u) = \sum_{i=1}^3 b_i e_i, \quad u = \sum_{i=1}^3 u_i e_i$$

By bilinearity and antisymmetry of the wedge product, this substitution in coordinates yields

$$\tau_{x,u} = \begin{pmatrix} 0 \\ e_1(x,u) \end{pmatrix} \wedge \begin{pmatrix} 0 \\ e_2(x,u) \end{pmatrix} = \sum_{i,j=1}^3 a_i b_j \begin{pmatrix} 0 \\ e_i \end{pmatrix} \wedge \begin{pmatrix} 0 \\ e_j \end{pmatrix} = \sum_{i<j,j=2}^3 (a_i b_j - a_j b_i) \begin{pmatrix} 0 \\ e_i \end{pmatrix} \wedge \begin{pmatrix} 0 \\ e_j \end{pmatrix}$$

The cyclic coefficients $(a_i b_j - a_j b_i)$ are nothing but the components of the cross product $e_1(x,u) \times e_2(x,u)$. As $(e_1(x,u), e_2(x,u), u)$ is a positively oriented orthonormal basis, it implies $u = e_1(x,u) \times e_2(x,u)$. This leads to the following expression

$$\tau_{x,u} = u_1 \begin{pmatrix} 0 \\ e_2 \end{pmatrix} \wedge \begin{pmatrix} 0 \\ e_3 \end{pmatrix} + u_2 \begin{pmatrix} 0 \\ e_1 \end{pmatrix} \wedge \begin{pmatrix} 0 \\ e_3 \end{pmatrix} + u_3 \begin{pmatrix} 0 \\ e_1 \end{pmatrix} \wedge \begin{pmatrix} 0 \\ e_2 \end{pmatrix} := u_1 \tilde{b}_1 + u_2 \tilde{b}_2 + u_3 \tilde{b}_3,$$

and subsequently,

$$\int_S \tau_{x,u} d\mathcal{H}^2(u) = \left(\int_S u_1 d\mathcal{H}^2(u) \right) \tilde{b}_1 + \left(\int_S u_2 d\mathcal{H}^2(u) \right) \tilde{b}_2 + \left(\int_S u_3 d\mathcal{H}^2(u) \right) \tilde{b}_3.$$

The spherical unit normal integral is computed explicitly in [11] as

$$\int_{S_\alpha^+} u d\mathcal{H}^2(u) = \pi \alpha, \quad \int_{\mathbb{S}^2} u d\mathcal{H}^2(u) = 0,$$

so that $\mathcal{C} = 0$ when S is the entire sphere. By substituting the previous expression into eq. (15), the spherical component of the triangulation reduces to,

$$N(\mathcal{T})^{sph} = -\pi \sum_{x_k \in \partial\mathcal{T}} \delta_{x_k} \alpha_k,$$

where one defines

$$\alpha_k := \sum_{f_i \text{ edge of } x_k} \sum_{j=1}^3 \frac{f_{ij}}{|f_i|} \tilde{b}_j$$

3.2.4 Dirac delta form

The Dirac delta form of the individual components may be substituted into eq. (11) in order to obtain

$$N(\mathcal{T}) = \pi \sum_{j=1}^{n_e} \delta_{c_j} \gamma_j + \pi \sum_{x_i \in \partial\mathcal{T}} \delta_{x_i} \alpha_i \quad (16)$$

as an element of W^* , where one defines

$$\alpha_i := - \sum_{f_i \text{ edge of } x_k} \sum_{j=1}^3 \frac{f_{ij}}{|f_i|} \tilde{b}_j$$

$$\gamma_i := - \sum_{T \text{ triangle of } e_j} \begin{pmatrix} f_j \\ |f_j| \\ 0 \end{pmatrix} \wedge \begin{pmatrix} 0 \\ f_j \times \beta_T \end{pmatrix}.$$

Given two triangulations $\mathcal{T}, \mathcal{T}'$, one may check the resulting scalar product of $N(\mathcal{T}), N(\mathcal{T}') \in W^*$ of eq. (16) agrees with eq. (9) derived in [11].

3.2.5 Real-valued embedding

With the Dirac delta decomposition eq. (16), we are almost in position to apply the compression algorithm of [9], to obtain a sparse approximation to $N(\mathcal{T}) \in W^*$. However, the compression algorithm assumes that W is an RKHS of \mathbb{R}^k valued functions for some $k \in \mathbb{N}$. To overcome this, one may identify the RKHS W of differential forms, with an RKHS V of \mathbb{R}^k valued functions. This type of embedding arises in [2] for currents and varifolds, where V is an RKHS of vector fields and real-valued functions respectively. We also show how one may compute this embedding efficiently for a given triangulation and associated normal-cycle.

The identification proceeds, by first noting the standard finite dimensional isomorphism $\Lambda^2(\mathbb{R}^n) \simeq \mathbb{R}^{\binom{n}{2}}$. We fix a natural choice of isomorphism $B : \Lambda^2(\mathbb{R}^n) \rightarrow \mathbb{R}^{\binom{n}{2}}$ induced by the standard basis $\tilde{\mathcal{B}} = \{e_1, \dots, e_n\}$ of \mathbb{R}^n , in the following way. Given a wedge product $\alpha = a \wedge b \in \Lambda^2(\mathbb{R}^n)$, we may expand in terms of $\tilde{\mathcal{B}}$ to obtain

$$\alpha = a \wedge b = \left(\sum_{i=1}^n a_i e_i \right) \wedge \left(\sum_{j=1}^n b_j e_j \right) = \sum_{i,j=1}^n a_i b_j (e_i \wedge e_j) \quad (17)$$

$$= \sum_{i < j, j=2}^n (a_i b_j - a_j b_i) (e_i \wedge e_j) := \sum_{k=1}^{\binom{n}{2}} c_k(a, b) b_k. \quad (18)$$

where we have used the bilinearity and antisymmetry properties of wedge product. In eq. (18), we have labelled the wedge basis

$$\mathcal{B} := \{b_k\}_{k=1}^{\binom{n}{2}} := \{e_i \wedge e_j\}_{i < j, j=2, \dots, n}.$$

The coefficients $c_k(a, b)$ are the cyclic coefficients in eq. (18) of the k 'th wedge basis element. With respect to the standard bases $\mathcal{B}, \tilde{\mathcal{B}}$, B may be constructed as the linear basis map such that $B(b_k) = e_k$, acting linearly on general wedge products as

$$B(a \wedge b) = (c_k(a, b))_{k=1}^{\binom{n}{2}} \in \mathbb{R}^{\binom{n}{2}}$$

As B is a linear map preserving orthonormal bases of the two Euclidean spaces, it is an isometric isomorphism.

Having fixed B , for surfaces — with $n = 6$ and $\binom{n}{2} = 15$ — we may isometrically identify the space W of differential forms $f : \mathbb{R}^3 \times \mathbb{S}^{d-1} \rightarrow \Lambda^2(\mathbb{R}^3 \times \mathbb{S}^{d-1})$, with an RKHS V of functions $f : \mathbb{R}^3 \times \mathbb{S}^{d-1} \rightarrow \mathbb{R}^{15}$, induced by the kernel

$$K((x, u), (y, v)) = K_p(x, y) Id_{15}, \quad \forall (x, u), (y, v) \in \mathbb{R}^d \times \mathbb{S}^{d-1}.$$

This in turn induces an isometric isomorphism of dual spaces $F : W^* \rightarrow V^*$, with the action $F(\delta_x \tau) = \delta_x B(\tau)$ on basis elements. As such, one may embed $N(\mathcal{T})$ into V^* as

$$F(N(\mathcal{T})) = \pi \sum_{i=1}^{n_e} \delta_{x_i} B(\gamma_i) + \frac{\pi}{2} \sum_{x_i \in \partial \mathcal{T}} \delta_{x_i} B(\alpha_i) \in V^* \quad (19)$$

with coefficient vectors in \mathbb{R}^{15} . Given the weights γ_i, α_i associated to $N(\mathcal{T})$, the weights of $F(N(\mathcal{T}))$ can be efficiently computed in $\mathcal{O}(n_e)$. With the embedding eq. (19), one is now in a position to apply the compression algorithm of [9] to approximate $F(N(\mathcal{T})) \in V^*$. Since F is an isometric isomorphism, forming approximations to $F(N(\mathcal{T}))$ in V^* will yield approximations of the same quality to $N(\mathcal{T}) \in W^*$, by applying F^{-1} .

3.3 Compression of Normal-Cycles

In the previous section, we have decomposed the Normal-Cycle embedding associated to a surface triangulation \mathcal{T} , or discrete curve \mathcal{C} , in a Dirac delta basis with \mathbb{R}^k valued weights. This expression has the generic form

$$\mu = \sum_{i=1}^n \delta_{x_i} \alpha_i \in V^*, \quad x_i \in \mathbb{R}^d, \quad \alpha_i \in \mathbb{R}^k,$$

where V is an RKHS induced by spatial kernel K_p . One may apply the compression algorithm of [9] in order to form a randomized approximation to μ of the form

$$\hat{\mu} = \sum_{i=1}^m \delta_{c_i} \beta_i \in V^*, \quad c_i \in \mathbb{R}^d, \quad \beta_i \in \mathbb{R}^k,$$

with $m \ll n$, c_i sampled from a μ dependent distribution and the weights depending on c_i and μ . The algorithm is given below

Algorithm 1 Discrete functional compression with RLS sampling

- 1: Fix domain \mathcal{X} , $m \ll n$, RKHS V , RKHS kernel function $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ and target functional

$$\mu = \sum_{i=1}^n \delta_{x_i} \alpha_i \in V^*, \quad (x_i, \alpha_i) \in \mathcal{X} \times \mathbb{R}^k.$$

- 2: Sample m distinct control points $\{c_1, \dots, c_m\} \subset \{x_1, \dots, x_n\}$ using Ridge Leverage Score (RLS) sampling.
- 3: Compute the vectors

$$y_j = \sum_{i=1}^n k(c_j, x_i) \alpha_i \in \mathbb{R}^k, \quad j = 1, \dots, m.$$

- 4: Form the control point approximation via orthogonal projection

$$\hat{\mu} = \sum_{i=1}^m \delta_{c_i} \beta_i \in V^*, \quad \beta = K_{CC}^{-1} y \in \mathbb{R}^{m \times k}.$$

Control points sampling using approximate Ridge leverage score sampling algorithm of [8, 4] is fast and yields strong theoretical bounds for decay of the compression error, as proven in [9].

Theorem 3. *Suppose we have a discrete target functional of the form*

$$\mu_S = \sum_{i=1}^n \delta_{x_i} \alpha_i \in V^*, \quad (x_i, \alpha_i) \in \mathcal{X} \times \mathbb{R}^k$$

with associated dual vector-valued function

$$v^\alpha(x) = \sum_{i=1}^n k(x, x_i) \alpha_i \in V.$$

Sample m distinct control points $\mathbf{c} = \{c_1, \dots, c_m\} \subset \{x_1, \dots, x_n\}$ and define the matrix of values

$$Y_c = (v^\alpha(c_1), \dots, v^\alpha(c_m))^T \in \mathbb{R}^{m \times k},$$

which is the evaluation of the dual vector-valued function on the control point locations. Computing weights

$$\beta = [\beta_1, \dots, \beta_m]^T = K_{cc}^{-1} Y_c \in \mathbb{R}^{m \times k},$$

yields an approximation

$$\sum_{i=1}^m \delta_{c_i} \beta_i \approx \sum_{i=1}^n \delta_{x_i} \hat{\alpha}_i,$$

that satisfies

$$\left\| \sum_{i=1}^n \delta_{x_i} \hat{\alpha}_i - \sum_{i=1}^m \delta_{c_i} \beta_i \right\|_{V^*}^2 \leq \text{Ctr}(K_{XX} - Q_{XX}), \quad Q_{XX} := K_{XC} K_{CC}^{-1} K_{CX}. \quad (20)$$

By sampling control points using the Recursive RLS scheme of [8], we obtain we obtain the following theoretical guarantees on the compress error. Fixing $\delta \in (0, \frac{1}{32})$, $S \in \mathbb{N}$, with probability $1 - 3\delta$, we have for $m \approx S \log(S/\delta)$ that

$$\left\| \sum_{i=1}^n \delta_{x_i} \hat{\alpha}_i - \sum_{i=1}^m \delta_{c_i} \beta_i \right\|_{V^*}^2 \leq \frac{Cn}{S} \sum_{i=S+1}^n \lambda_i(K_{XX}). \quad (21)$$

That is, the randomized compression error decays at least as fast as the decay of sum of eigenvalues of the spatial kernel matrix, evaluated on the delta centres. For typical choices of the spatial kernel such as Gaussian RBF, this decay is exponential in m [9], and allows one to compress the target normal cycle to a very small $m \ll n$, with very small error in the RKHS norm. We demonstrate this in the following sections on extensive numerical examples, as well as large-scale nonlinear registration problems.

4 Numerical Experiments

We now demonstrate the effectiveness of algorithm 1 for compression of large-scale normal-cycles, on surfaces from modern geometry processing datasets. We demonstrate properties such as rapid approximation error decay and effectiveness in down-line applications to nonlinear LDDMM registration.

4.1 Compression error decay

In this section, we study rate of decay of the compression error in RKHS norm for normal-cycles as a function of m , the number of samples. We use the following surface data for our experiments in this section.

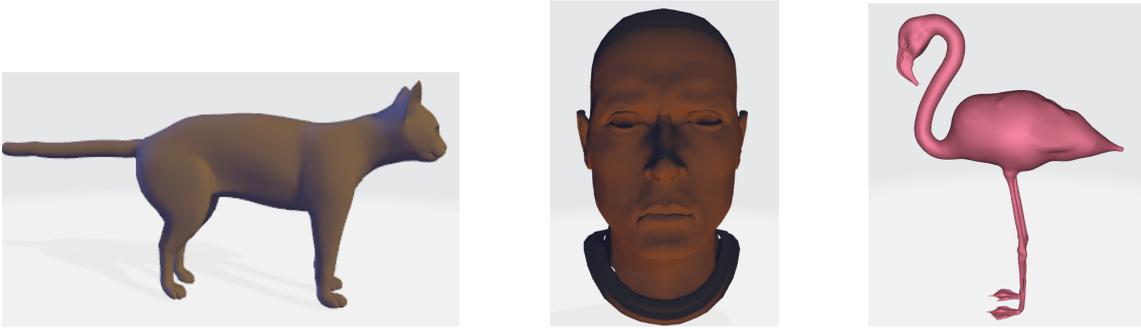


Figure 1: **Left:** Cat (14410 triangles) **Middle:** Head (31620 triangles) **Right:** Flamingo (52895 triangles)

The data are centred and scaled, so that the flamingo surface lies in a box of size $1.3 \times 3.3 \times 7.1$, the head surface lies in a box of size $3.8 \times 5.3 \times 4.0$ and the cat surface lies in a box of size $1.5 \times 5.3 \times 3.8$. We run algorithm 1, on our test surfaces. For each test surface, we calculate and plot the true square error in RKHS norm of the RLS compression as a function of m . We also plot for comparison, the error curve for Uniform sampling, where control points are simply subsampled according to a discrete uniform distribution. Finally, the trace bound on the squared error derived in Corollary ?? is also plotted, with rescaled numerically tighter constants. Note that we do not plot the curves for eigenvalue bounds eq. (21), due to the prohibitive cost of computing the eigen-decomposition for large n . For the spatial kernel, we make the choice of Gaussian RBF kernel, $k(x, y) = \exp(-\frac{\|x-y\|^2}{2\sigma^2})$, with scale parameter $\sigma_p \in \{0.3, 0.5, 0.25\}$ respectively, for the cat, head and flamingo test cases. The resulting plots are shown in fig. 2.

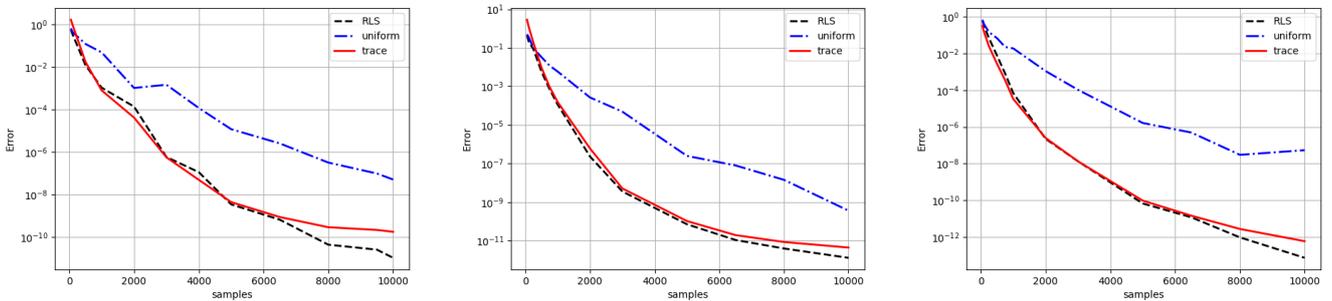


Figure 2: Numerical curves comparing RKHS error decay (black) of RLS compression of normal-cycles, to theoretical trace bound (red) and Uniformly sampled compression (blue), on cat (left), head (centre) and flamingo (right) surfaces.

We observe that the decay of error of the compressed approximation is rapid, and one can take $m \ll n$ for a good quality of approximation, across all the example cases. One also observes, that the theoretical trace bounds (in red), have decay rate that generally matches that of the true squared error. We also note that the speed of compression is rapid; even for the large flamingo test case, the compression to 1000 delta centres ($< 1\%$ error and around 99% compression ratio) takes at most 1 second.

Finally, we observe that RLS sampling consistently outperforms uniform sampling in all cases. The RLS sampling will also produce better *quality* samples (in terms of Nystrom error) than uniform sampling. In general, uniform sampling will tend to have more points represented in regions which are over-sampled to begin with, thus only representing the underlying surface well only in densely sampled regions. On the other hand, RLS sampling measures the local ‘importance’ of triangles through the ridge leverage scores and samples them accordingly. Thus, RLS sampling tends to

produce more ‘diverse’ samples [8] and will produce samples that are well spread out independent of the initial sampling density. This is especially advantageous for compressing real-world geometries, which have varying resolution across different geometric locations.

Finally, we note that in many geometry processing applications, one often does not require a machine precision level error in order to perform down-line tasks with the compressed representation. In many situations, the data itself is often acquired in a noisy way, and can contain many local variations that are not relevant in describing the global geometry. As such, the above curves suggest one can practically choose $m \ll n$ and obtain an acceptable error (within the data uncertainty) for the down-line tasks which we perform with the compression. For example, [5] suggests a heuristic of $\tau = 5\%$ relative error cut-off for compression of currents, which is also a reasonable approach for normal-cycle compression.

4.2 Matching quality

Finally, we illustrate the effectiveness of the compressed normal-cycles for nonlinear registration in the LDDMM framework. We demonstrate that one may obtain high quality of registration using only the compressed representations with $m \ll n$, even when only 1 – 2% of the underlying triangles are used in the compression. We shall also measure the massive computational savings of the compressed registration algorithm, in terms of memory and run-time.

We demonstrate on two extremely densely sampled shapes, taken from modern geometry processing datasets. The first, is the ‘PumpkinHead’ test surface with 394,510 triangles, and the second the ‘Queen’ test surface taken from the Thingi10K dataset with 229,776 triangles. In the full normal-cycle representation, the PumpkinHead surface has 591,765 Dirac delta centres, and the Queen test surface has 344,664 Dirac delta centres. The PumpkinHead surface is centred and lies in a box of size $3.5 \times 3.8 \times 3.9$, and the Queen test surface in a box of size $3.1 \times 4.0 \times 4.9$. In both cases, we consider a Hamiltonian LDDMM matching problem ???. We use the normal-cycles metric as discrepancy, induced by Gaussian kernel with parameters $\sigma_p = 0.2$ and a spherical template. For the spatial kernel K_V parametrizing vector fields in the LDDMM framework, we fix a sum of 4 Gaussian kernels of decreasing length-scales $\sigma_i \in \{1.0, 0.5, 0.2, 0.1\}$.

With this problem configuration, in both test cases we compare the matching quality and runtime for the uncompressed, and *RLS compressed matching algorithm* of [9] adapted for compressed normal cycles. The matching quality is measured in the Hausdorff metric over sets given as

$$d_H(A, B); = \max\left(\sup_{a \in A} d(a, B), \sup_{b \in B} d(b, A)\right), \quad d(x, A) := \inf_{a \in A} \|x - a\|_2,$$

which is independent of the compression and matching algorithm. For all experiments, the diffeomorphism and push-forward are computed via a forward Euler scheme with 5 time-steps. The kernel and gradient computations for the diffeomorphisms, are performed using Keops and automatic differentiation. In the Pumpkinhead surface test case, we use 8000 control points to parametrize the diffeomorphism. In the Queen test surface case, we use 10000 control points to parametrize the diffeomorphism. Optimization is performed via an LBFGS routine, with strong-wolfe line search run for 500 iterations. All experiments are performed on a Tesla T4 GPU with 16gb of ram.

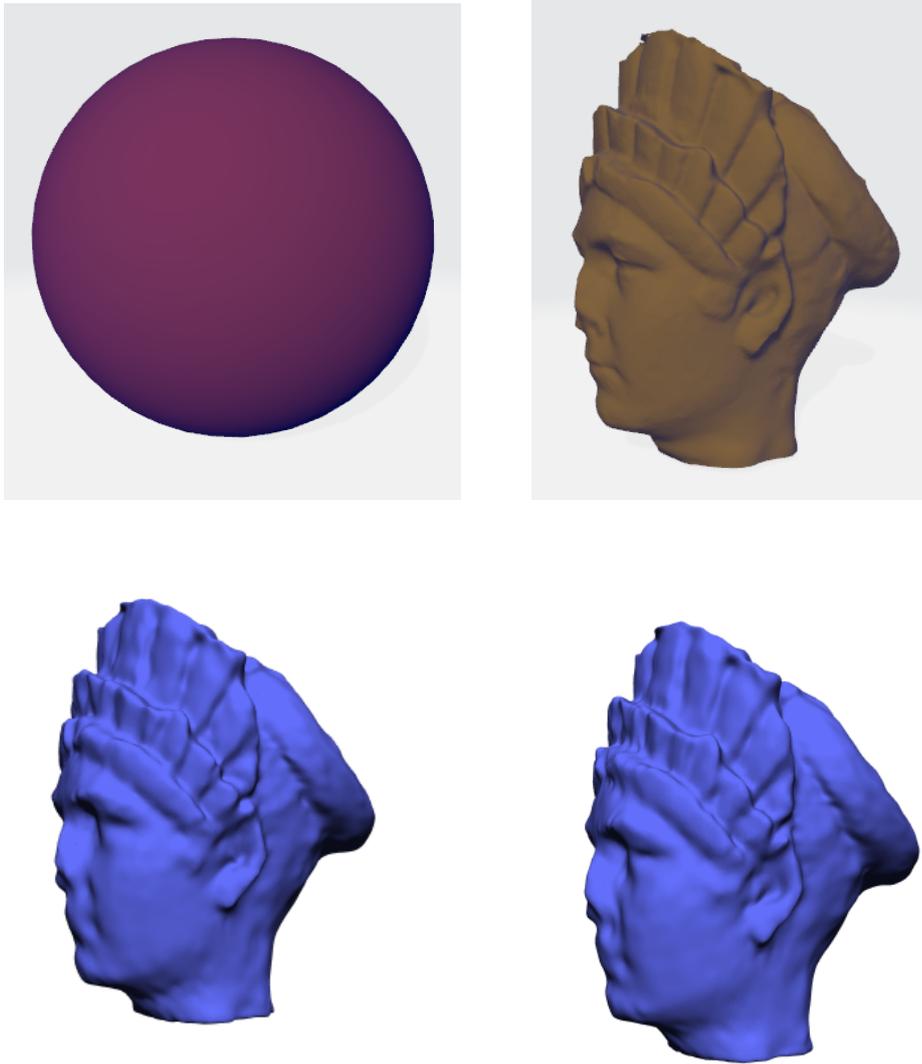


Figure 3: **Top left:** spherical template. **Top right:** target mesh. **Bottom left:** Matching with full metrics taking 2 hours and 42 minutes with $d_H = 0.0442$. **Bottom Right:** Matching with 97% compression of template and target taking only 17 minutes with $d_H = 0.0298$.

For the ‘queen’ surface test case, we compress the target and template down to 10,000 Dirac delta each, from the original 344,664 Dirac delta centres, which is a compression ratio of over 97%, giving a significant memory saving of the resulting matching algorithm. We observe in fig. 3, that the matching quality is almost identical between compressed and uncompressed and in some regions better than the full matching. This is numerically evidenced by the near identical Hausdorff metric score. As one expects, the compressed matching algorithm yields a significant speedup of 9 – 10 times over the uncompressed version, reducing overall matching time from 9720s to 1020s.

We now move onto the second test case on the ‘pumpkinhead’ test surface with 350,328 triangles, and 591,765 Dirac centres in the normal cycle. The data and results are shown in fig. 4.



Figure 4: **Top left:** spherical template. **Top right:** target mesh. **Bottom left:** an example matching with full normal-cycles, from spherical template to target without compression, taking 5 hours and 37 minutes with $d_H = 0.1621$. **Bottom right:** the same example matching but with 99% compression taking only 17 minutes with $d_H = 0.1662$.

We compress the target and template down to 5000 Dirac delta centres each, from the original full resolution normal-cycle with 591,765 Dirac centres, yielding a compression ratio of over 99%. Once again, this yields huge memory savings and on this large-scale example yields almost a factor of 20 times speed-up over the uncompressed matching problem, reducing overall matching time from 20220s to 1020s, on 500 iterations. We observe in fig. 4 that a comparable quality of matching is achieved even with this level of compression, which is also reflected in the Hausdorff metric score.

5 Conclusion

In this work, we have extended the algorithm of [9] to compression of normal-cycles, allowing us to scale up the normal-cycles metric to shapes with extremely dense resolution of the order $10^5 - 10^6$, while keeping the computational cost at a fraction of the full representation. Furthermore, our compression comes with fast error decay, and guarantees of small RKHS distance to the normal cycle of the true underlying shape. We have demonstrated the effectiveness of the compression method on real-world shape data in downline tasks such as nonlinear LDDMM registration. Indeed, our application of this method to nonlinear LDDMM registration shows vastly superior run-times, with negligible difference in registration quality. We leave as future work, the extension the compression algorithm to the linear and Gaussian normal kernel case, which can help increase the sensitivity of the underlying shape representations, to higher order curvature information.

References

- [1] B. CHARLIER, J. FEYDY, J. A. GLAUNÈS, F.-D. COLLIN, AND G. DURIF, *Kernel operations on the gpu, with autodiff, without memory overflows*, Journal of Machine Learning Research, 22 (2021), pp. 1–6.
- [2] N. CHARON, B. CHARLIER, J. GLAUNÈS, P. GORI, AND P. ROUSSILLON, *12 - fidelity metrics between curves and surfaces: currents, varifolds, and normal cycles*, in Riemannian Geometric Statistics in Medical Image Analysis, X. Pennec, S. Sommer, and T. Fletcher, eds., Academic Press, 2020, pp. 441–477.
- [3] N. CHARON AND A. TROUVÉ, *The varifold representation of nonoriented shapes for diffeomorphic registration*, SIAM journal on imaging sciences, 6 (2013), pp. 2547–2580.
- [4] F. CHERFAOUI, H. KADRI, AND L. RALAIVOLA, *Scalable ridge leverage score sampling for the nystrom method*, in ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2022, pp. 4163–4167.
- [5] S. DURLEMAN, X. PENNEC, A. TROUVÉ, AND N. AYACHE, *Statistical models of sets of curves and surfaces based on currents*, Medical Image Analysis, 13 (2009), pp. 793–808. Includes Special Section on the 12th International Conference on Medical Imaging and Computer Assisted Intervention.
- [6] H. FEDERER, *Geometric measure theory*, Classics in Mathematics ; v.153, Springer-Verlag, Berlin, Heidelberg, 1996.
- [7] H.-W. HSIEH AND N. CHARON, *Metrics, quantization and registration in varifold spaces*, Foundations of computational mathematics, 21 (2021), pp. 1317–1361.
- [8] C. MUSCO AND C. MUSCO, *Recursive sampling for the nystrom method*, in Advances in Neural Information Processing Systems, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, eds., vol. 30, Curran Associates, Inc., 2017.
- [9] A. PAUL, N. CAMPBELL, AND T. SHARDLOW, *Sparse nystrom approximation of currents and varifolds*, 2024.
- [10] P. ROUSSILLON, *Normal cycle models for deformation analysis*, theses, Université Sorbonne Paris Cité, Nov. 2017.
- [11] P. ROUSSILLON AND J. A. GLAUNÈS, *Representation of surfaces with normal cycles and application to surface registration*, Journal of mathematical imaging and vision, 61 (2019), pp. 1069–1095.
- [12] ———, *Representation of surfaces with normal cycles and application to surface registration*, Journal of mathematical imaging and vision, 61 (2019), pp. 1069–1095.
- [13] M. VAILLANT AND J. GLAUNÈS, *Surface matching via currents*, INFORMATION PROCESSING IN MEDICAL IMAGING, PROCEEDINGS, 3565 (2005), pp. 381–392.
- [14] L. YOUNES, *Shapes and Diffeomorphisms*, vol. 171 of Applied Mathematical Sciences, Springer Berlin / Heidelberg, Berlin, Heidelberg, 2010.

A Exterior algebra and Differential forms

We introduce necessary background on exterior algebra required for defining Normal Cycles, and define notions such as wedge product and differential forms, in terms of alternating multilinear maps.

A.1 Exterior algebra

Given a vector space V and two elements $v, w \in V$, the wedge product is an operation on v, w which gives a way to ‘multiply’ vectors to obtain a vector $v \wedge w$ in a new space (the exterior algebra), with the properties that,

- \wedge is bilinear. That is, for $\alpha \in \mathbb{R}$, $x, y, z \in V$ we have the relation

$$\begin{aligned}\wedge(x + \alpha y, z) &= \wedge(x, z) + \alpha \wedge(y, z), \\ \wedge(x, y + \alpha z) &= \wedge(x, y) + \alpha \wedge(x, z),\end{aligned}$$

- \wedge is antisymmetric, so that for $x, y \in V$,

$$\wedge(x, y) = -\wedge(y, x)$$

- \wedge is alternating so that for $x, y \in V$,

$$\wedge(x, x) = 0$$

The wedge product of two vectors contains all the information about the oriented plane spanned by x, y , and has modulus equal to the area of the associated parallelogram. The space in which $x \wedge y$ lives is denoted $\Lambda^2 V$, the 2nd exterior power of V , which consists of formal linear combinations of elements of the form $x \wedge y$, giving it a vector space structure. One can also wedge together vectors in V and $\Lambda^2 V$, to obtain analogously $\Lambda^3 V$, the third exterior power of V consisting of formal linear combinations of $v \wedge w \wedge g$. This process can be repeated to obtain the k 'th exterior power spaces $\Lambda^k V$ for all $k \in \mathbb{N}$, and taking the direct sum of these spaces yields the *exterior algebra*

$$\bigwedge V := \bigoplus_{k=0}^{+\infty} \Lambda^k V,$$

which is a vector space consisting of all formal linear combinations of varying sizes, of wedge products. The exterior algebra is indeed an algebra under the product operation \wedge .

The formal, fully rigorous construction of the exterior algebra is performed via taking quotients of the tensor algebra $T(V)$ associated to the base vector space, under an equivalence relation defined by the properties given above. Instead for our purposes, we give a simpler more concrete definition, following [5]. We begin by defining the space of alternating multilinear forms on a vector space V .

Definition A.1. Given a vector space V , an alternating multilinear form of degree k , is a mapping $f : V \times \dots \times V \rightarrow \mathbb{R}$ with the properties that,

- f is linear in each entry.
- $f(v_1, \dots, v_n) = 0$, whenever v_1, \dots, v_n are linearly dependent.

We denote the space of such mappings as $A^k V$ which is a vector space under pointwise addition and multiplication.

We now define the k 'th exterior product space $\Lambda^k V$ for *finite dimensional* V as follows.

Definition A.2. Given a finite dimensional vector space V , we define the k 'th exterior product space as $\Lambda^k V := (A^k V)^*$, the dual space of $A^k V$.

Note also that for finite dimensional V , both these spaces are finite dimensional, as multilinear forms are entirely determined by their values on the basis tuples, and therefore it's dual is also finite dimensional.

One now defines the exterior product of vectors $v_1, \dots, v_k \in V$ as the element denoted $v_1 \wedge \dots \wedge v_k \in \Lambda^k V$ as,

$$(v_1 \wedge \dots \wedge v_k)(f) := f(v_1, \dots, v_k), \quad f \in A^k V$$

It can be shown that such products form a basis of $\Lambda^k V$, so that the space is made up on linear combinations of the exterior product and satisfies $\dim(\Lambda^k V) = \binom{\dim(V)}{k}$. Furthermore, it can be verified the wedge as defined above is linear in each variable, and vanishes when v_1, \dots, v_k are linearly dependent. With this definition, one can also identify $A^k V \cong (A^k(V))^{**} \cong (\Lambda^k V)^*$, with the action,

$$f(v_1 \wedge \dots \wedge v_k) := (v_1 \wedge \dots \wedge v_k)(f) = f(v_1, \dots, v_k)$$

Having defined the exterior power spaces, one may extend the wedge as follows,

$$\begin{aligned} \wedge : \Lambda^k V \times \Lambda^l V &\longrightarrow \Lambda^{k+l} V \\ (v_1 \wedge \dots \wedge v_k) \wedge (w_1 \wedge \dots \wedge w_l) &:= (v_1 \wedge \dots \wedge v_k \wedge w_1 \wedge \dots \wedge w_l) \end{aligned}$$

where once again one can show is associative, bilinear and anticommutative.

Finally, one may define the exterior algebra as $\bigwedge V := \bigoplus_{k=0}^{+\infty} \Lambda^k V$, and verify that this is an algebra equipped with formal addition and \wedge as the product. This approach agrees with the more technical construction via quotients mentioned previously, up to isomorphism.

One may also define an inner product on the exterior power spaces $\Lambda^k V$ by defining it on simple wedges of k vectors. In particular, one can define an inner product via the inner product in the basis induced by standard basis of \mathbb{R}^d , which agrees with the following coordinate independent formula,

$$\langle v_1 \wedge \dots \wedge v_k, w_1 \wedge \dots \wedge w_k \rangle_{(\Lambda^m \mathbb{R}^n)} := \det(\langle v_i, w_j \rangle)_{1 \leq i, j \leq k}$$

A.2 Differential Forms

Having defined the wedge product on \mathbb{R}^n and m -forms, one may define differential forms. A differential m form on \mathbb{R}^n for $m \leq n$, is defined as a smooth assignment of m -form on \mathbb{R}^n . For the purposes of Geometric measure theory and constructing the normal-cycle of a surface, one usually considers compactly supported differential forms, so that one restricts to $\omega \in C_0(\mathbb{R}^n, (\Lambda^m \mathbb{R}^n)^*)$. One places the infinity norm on this space, so that

$$\|\omega\|_\infty = \sup_{x \in \mathbb{R}^n} |\omega(x)|_{(\Lambda^m \mathbb{R}^n)^*}.$$

An m form on \mathbb{R}^n , is the natural object that one can integrate over submanifolds, and leads to a coordinate independent notion of integration on manifolds [5].

B Rectifiable sets and Hausdorff measure

The notion of Hausdorff measure and rectifiable sets are a key ingredient in describing the normal-cycle associated to a shape, and so we review these concepts in the following section.

B.1 Hausdorff measure

The m -dimensional Hausdorff measure on \mathbb{R}^n , is defined as follows. First, one defines

$$\mathcal{H}_\delta^m(S) := \inf \left\{ \sum_{i=1}^{\infty} \text{diam}(U_i)^d \mid U_i \text{ open, } S \subset \cup_{i=1}^{\infty} U_i, \quad \text{diam}(U_i) < \delta \right\},$$

which is the effective ‘size’ of the smallest cover by open sets with diameter bounded by δ . One takes the limit as the size of cover $\delta \rightarrow 0$, to obtain the outer measure

$$\mathcal{H}^m(S) = \lim_{\delta \rightarrow 0} \mathcal{H}_\delta^m(S).$$

which can be made into a measure by restricting to measurable sets with respect \mathcal{H}^m . The resulting measure is m -dimensional Hausdorff measure on \mathbb{R}^n . When $m = n$, this agrees with λ^n the Lebesgue measure in \mathbb{R}^n . This is a natural measure to use for integration and measuring ‘volume’ on submanifolds embedded in \mathbb{R}^n , as the Hausdorff measure $\mathcal{H}^m(S)$ agrees with the intuitive notion of volume for submanifolds S of dimension m embedded in \mathbb{R}^n , where the Lebesgue measure would give measure 0. It is also useful for defining integration of forms on non-smooth subsets, such as rectifiable sets which we now discuss.

B.2 Rectifiable sets

Rectifiable sets are a generalization of smooth manifolds, and are the typical objects which are studied in Geometric measure theory. Familiar objects such as smooth manifolds are rectifiable. Piece-wise smooth sets, such as triangulated surfaces are also rectifiable, as well as much rougher Cantor-set like objects.

Definition B.1. An m -rectifiable set, is $X \subset \mathbb{R}^d$ ($m \leq d$) is one which can be covered \mathcal{H}^m almost everywhere, by the union of images of Lipschitz functions, on bounded sets. That is, there exists bounded $U_i \subset \mathbb{R}^m$ and Lipschitz functions $f_i : U_i \subset \mathbb{R}^m \rightarrow \mathbb{R}^d$, such that

$$\mathcal{H}^m(X - C) = 0, \quad C := \cup_{i=1}^{\infty} f(U_i).$$

Unlike the standard case of smooth manifolds, the charts for the set X are not necessarily smooth, allowing X to have corners and singularities. Rectifiable subsets despite their general definition, can be shown to have well defined tangent spaces \mathcal{H}^m almost everywhere [12] which allows one to generalize familiar notions of differential geometry on smooth manifolds in a measure theoretic sense, such as integration of smooth differential forms over rectifiable sets. This leads to the notion of rectifiable currents which is covered in section 2.