We present a comprehensive analysis of total variation (TV) on non-Euclidean domains and its eigenfunctions. We specifically address parameterized surfaces, a natural representation of the shapes used in 3D graphics. Our work sheds new light on the celebrated Beltrami and Anisotropic TV flows and explains experimental findings from recent years on shape spectral TV [Fumero et al. 2020] and adaptive anisotropic spectral TV [Biton and Gilboa 2022]. A new notion of convexity on surfaces is derived by characterizing structures that are stable throughout the TV flow, performed on surfaces. We establish and numerically demonstrate quantitative relationships between TV, area, eigenvalue, and eigenfunctions of the TV operator on surfaces. Moreover, we expand the shape spectral TV toolkit to include zero-homogeneous flows, leading to efficient and versatile shape processing methods. These methods are exemplified through applications in smoothing, enhancement, and exaggeration filters. We introduce a novel method that, for the first time, addresses the shape deformation task using TV. This deformation technique is characterized by the concentration of deformation along geometrical bottlenecks, shown to coincide with the discontinuities of eigenfunctions. Overall, our findings elucidate recent experimental observations in spectral TV, provide a diverse framework for shape filtering, and present the first TV-based approach to shape deformation.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM Trans. Graph., Vol. 43, No. 2, Article 22. Publication date: February 2024.

1 INTRODUCTION

Spectral geometry processing is a widely used technique in computer graphics. It involves breaking down shapes and their functions into spectral components, harnessing the multiscale nature and the well-studied properties of eigenfunctions [Aflalo et al. 2013; Bracha et al. 2020; Sorkine et al. 2004; Taubin 1995; Wetzler et al. 2013]. Spectral processing is useful for various computer graphics applications, such as mesh filtering, surface decoding, segmentation (spectral partitioning), shape deformation, and general analysis. For a comprehensive exploration of this approach, including a survey and overview, please refer to Cammarasana and Patané [2021] and the references therein.

Generally, the spectral approach is based on linear algebra and harmonic analysis. It can be viewed as a generalization of Fourier basis to Riemannian manifolds and to graphs. In recent years, there has been a growing branch studying nonlinear spectral decompositions, see, e.g., Gilboa [2018]. In this setting, a signal is decomposed into spectral components related to nonlinear eigenfunctions of the form $\lambda u \in \partial J(u)$, where $\partial J$ is the subdifferential of a convex functional $J$. It can be readily seen that when $J$ is the Dirichlet energy, $J(u) = \|\nabla u\|_2^2$, we have $\partial J(u) = \nabla J(u) = -\Delta u$, where $\Delta$ denotes the Laplacian. Thus, the eigen-decomposition reduces back to a linear Fourier-type analysis. In this sense, it is a generalization of linear spectral methods. Most of the research was dedicated to absolutely one-homogeneous functionals [Bungert et al. 2021; Burger et al. 2016; Gilboa 2014] such as total variation. These new signal analysis methods enabled crisp, edge-preserving decompositions and representations. However, the theory and applications were restricted mostly to images and signals in Euclidean domains. In this study, we develop a theoretical setting for nonlinear
spectral processing on surfaces, demonstrating the potential benefits of this approach to computer graphics.

**Total Variation (TV)** is a popular regularization function, well-known for its edge-preserving properties [Chambolle 2004]. For a smooth function \( u : \Omega \rightarrow \mathbb{R} \), the TV functional is

\[
TV(u) = \int_{\Omega} |\nabla u(x)|\,dx.
\]  

(1)

Total variation has found extensive applications in various domains and tasks over the past three decades. Most notably, this approach was extensively used in image processing, for denoising, deconvolution, texture modeling, super-resolution, segmentation, and more [Aujoil et al. 2006; Burger and Osher 2013; Chambolle and Pock 2011; Leng et al. 2021; Pascal et al. 2021; Rudin et al. 1992; Wei et al. 2019; Zhang et al. 2022].

The exploration of Total Variation extends beyond images and encompasses shapes, point clouds, and graphs [Dinesh et al. 2020; Duan et al. 2019; Sawant and Prabukumar 2020]. Consequently, it is natural to harness the TV functional for its nonlinear spectral properties in fields like spectral geometry. Such an approach was initiated by Fumero et al. [2020], which introduced Spectral Total Variation [Gilboa 2013, 2014] for geometry analysis. Nonetheless, transitioning from images to shapes presents new challenges.

In the domain of image processing, regularization predominantly revolves around manipulating a function \( u \) defined within a Euclidean domain. However, when it comes to geometry processing, the usage of \( u \) as a representation of the processed shape introduces a significant distinction, as the assumption of a Euclidean metric is typically not valid. For Spectral TV, this distinction sparks unique and unexplored considerations.

The primary focus of this article centers around the theory of TV eigenfunctions and Spectral TV within the realm of 3D geometry. Our main contributions are:

- A new way to generalize the definition of convex sets from planar domains to surfaces. This generalization arises as a property of total variation (TV) eigenfunctions on non-Euclidean surfaces, addressing theoretical inquiries of recent years [Fumero et al. 2020; Biton and Gilboa 2022].
- Further quantitative relationships between eigenvalue, area, and the total variation of the eigenfunctions are derived.
- Numerical demonstrations of our theoretical findings on the eigenfunction properties are provided in Figures 3, 4, 5.
- Spectral TV is extended to include novel zero-homogeneous flows, utilized for spectral filtering of shapes, where each flow induces distinct filtering capabilities—while being multi-scale as expected from spectral methods (see for instance Figures 8, 10).
- The discovered properties of eigenfunctions are observed in both our shape filtering and shape deformation results, enhancing the understanding of these methods, notably in Figures 5, 11, 12, 13.
- We present the first TV-based solution to the Shape Deformation task, as described in the tutorial by Sorkine and Botsch [2009]. This approach typically results in the concentration of deformation around geometrical bottlenecks (see Figures 14, 20).

Through these contributions, we not only derive new generalizations of key theoretical concepts from Euclidean planes to non-Euclidean surfaces but also showcase how these theoretical advancements can be applied to shape processing tasks, offering new insights as well as methodologies. For 3D model details, see footnote under References. Our results are reproducible via our official implementation.

2 PREVIOUS WORK

Shape filtering has a long and rich history. The pivotal work of Taubin [1995] proposed to utilize the shape-induced Laplacian eigenfunctions as a basis for shape filtering in an analogous manner to classical signal processing techniques. A transform is computed by projecting the shape onto the basis, where filtering is obtained by weighted reconstruction via this basis. Many variations of this method were utilized for different tasks (e.g., Sorkine et al. [2004]).

Over time, the Laplace-Beltrami became the standard Laplacian of choice for spectral applications and in general [Wetzler et al. 2013]. For non-rigid shape processing, spectral representations of a scale-invariant version of the Laplace-Beltrami was introduced in Aflalo et al. [2013] and later combined with deep learning for shape correspondence [Brach et al. 2020], providing a non-rigid spectral analogue of the popular spatial-domain approach of Litany et al. [2017].

While TV is traditionally used for image processing (see Burger and Osher [2013], Chambolle et al. [2010] for theory and applications), it was used in computer graphics as well for surface denoising, reconstruction, segmentation, and super-resolution [Dinesh et al. 2019, 2020; Kerault and Lachaud 2020; Liu et al. 2017; Zhang and Wang 2020; Zhang et al. 2015; Zhong et al. 2018]. Only recently, in 2020, a TV-based shape filtering approach was proposed [Fumero et al. 2020], advocating the application of spectral TV to the normals of the shape.

TV and shape smoothing flows have a substantial background, spanning a considerable duration. The authors of Kimmel et al. [1998] established a close relation between TV image smoothing and Laplace-Beltrami shape smoothing. These techniques were based on discrete diffusion processes. Among a plethora of Laplacian-based diffusion processes, we note two fundamental flows: the Mean Curvature Flow (MCF) and the Willmore flow, which gave rise to a new problem of singularities [Blatt 2009; Desbrun et al. 1999; Huiskes 1990]. More stable variations of them were proposed throughout the years, for instance, Crane et al. [2013] and Kazhdan et al. [2012]. In Emoza et al. [2008], a generalization of the Laplacian-based approaches was proposed. Using graph-oriented operators, they introduced p-Laplace flows for mesh fairing, including the 1-Laplace flow, which is a TV-flow, the gradient flow minimizing the TV energy.

Various methods have been proposed to minimize numerically TV and a \( L^2 \) square fidelity term \( (J_{TV}(u) + \lambda \|f - u\|_2^2) \), often referred to as the ROF model [Rudin et al. 1992]), see for instance Chambolle [2004]; Chambolle and Pock [2011]. For implementing
a TV-flow, one should note that the term $-\text{div} \left( \frac{\nabla u}{|\nabla u|} \right)$ is valid for the gradient estimation of the energy only for non-vanishing gradients of $u$. Two common approaches are adopted to overcome this. One is to use a regularized TV model $J_{TV}^{-\epsilon} := \int_{\Omega} \sqrt{\nabla u(x)^T \nabla u(x)} + \epsilon^2 \, dx$. In this case, one does not obtain singularities at zero gradients, and a straightforward explicit method can be used. Drawbacks of this approach are that an additional $\epsilon$ parameter is introduced and that the flow is smoother (somewhat less edge-preserving). Moreover, the evolution timestep is highly restrictive, proportional to $\epsilon$. Alternatively, one can approximate the gradient descent process as a series of proximal TV minimizations and use non-smooth solvers to obtain fully edge-preserving solutions, e.g., by Chambolle [2004] or Chambolle and Pock [2011]. This yields more faithful results. These methods can be regarded as semi-implicit methods, which are unconditionally stable (with respect to the timestep parameter, as opposed to explicit methods); see Weickert et al. [1998].

In our work, we adapt two such processes for geometry processing: the first is the shape flow introduced in Kazhdan et al. [2012], which proposed a gradient descent semi-implicit flow (see Parikh et al. [2014] for semi-implicit approaches). It is adapted by changing their proposed gradient direction to conform with our setting. The second is the iterative re-weighted L1 minimization scheme, introduced in Bronstein et al. [2016], which is adapted by using a vectorial version of it, and again, selecting our operator of choice to replace their gradient direction. The shape deformation problem is classically solved as a constrained energy-minimization problem, for instance, Botsch and Sorkine [2007]; Sorkine and Alexa [2007], which we also explore in our work.

Another approach that relates to our method is the family of shape processing techniques that process the displacement fields over a smooth version of the surface. This is mainly used for shape smoothing, exaggeration, and also for detail transfer, for instance, Cignoni et al. [2005]; Digne [2012]; Sorkine and Botsch [2009]. Recently, Yifan et al. [2021] leveraged two separate neural networks for the over-smoothed shape and for the displacement.

Our most significant contribution is to the theory of spectral TV. Let us recap important landmarks of this domain for the Euclidean setting. Spectral TV was introduced in Gilboa [2013, 2014], facilitating nonlinear edge-preserving image filtering. Essentially, the idea is to decompose a signal into nonlinear spectral elements related to eigenfunctions of the total-variation subdifferential. These nonlinear eigenfunctions raise important theoretical aspects—as their properties enable an understanding of the behavior of both TV regularization and Spectral Total Variation processing. The method is based on evolving gradient descent with respect to the TV functional, also known as the TV-flow [Andreu et al. 2001]. The spectral elements decay linearly in this flow. Different decay rates correspond to different scales, where in the case of a single eigenfunction the rate is exactly the eigenvalue. Theoretical underpinning was performed for the spatially discrete case in Burger et al. [2016], which also extended the concepts of spectral TV to decompositions based on general convex absolutely one-homogeneous functionals. Decompositions based on minimizations with the Euclidean norm, as well as with inverse-scale-space flows [Burger et al. 2006] were also proposed. The space-continuous setting was later analyzed in Bungert et al. [2021]. For the one-dimensional TV setting, it was shown that the spectral elements are orthogonal to each other. Various applications were suggested for image enhancement, manipulation and fusion [Benning et al. 2017; Hait and Gilboa 2019]. A common thread related to gradient flows of one-homogeneous functionals is that they are based on zero-homogeneous operators. Note that other homogeneities are also possible, see for instance Bungert and Burger [2020]; Cohen and Gilboa [2020].

While in non-Euclidean domains the theory of TV eigenfunctions and spectral TV is not as developed as in the Euclidean case, other aspects of TV on non-Euclidean domains are highly researched. A TV framework on manifolds was defined in Ben-Artzi and LeFloch [2007]. This research was in the context of nonlinear hyperbolic conservation laws on Riemannian manifolds. The authors prove bounds and stability of the minimizing flow. Another example is anisotropic TV, which is typically applied for image processing (see Grasmair and Lenzen [2010]) and can be analyzed as a functional on a non-Euclidean domain—as shown in Biton and Gilboa [2022], which conducted experiments with spectral TV as well. In Fumero et al. [2020] experiments with geodesically convex sets were conducted. In their experiments, geodesically convex sets exhibited a linear decay throughout the minimizing flow. Theoretical explanations of this phenomenon were left as open research questions. In the following, we lay theoretical foundations for these experimental results, validated by our proposed spectral nonlinear and non-Euclidean framework.

### 3 Preliminaries

In this work, the processed shape is assumed to be a 2D manifold $M \subset \mathbb{R}^3$. We assume $M$ is smooth, with a smooth parameterization $S : (\omega_1, \omega_2) = (x(\omega_1, \omega_2), y(\omega_1, \omega_2), z(\omega_1, \omega_2)), \omega_1, \omega_2 \in \Omega$, i.e.,

$$S : \Omega \subset \mathbb{R}^2 \to M.$$  

(2)

Namely, $S$ is differentiable and invertible. In the following, we outline important well-known properties of this setting, which we use in our work. A celebrated resource for these properties is Do Carmo [2016].

Let $f : M \to \mathbb{R}$, and $u = f \circ S(\omega_1, \omega_2)$, i.e., $u : \Omega \to \mathbb{R}$. Let $T_q M$ be the plane tangent to $M$ at point $S(\omega_1, \omega_2) = q \in M$. It can be shown that $\frac{\partial}{\partial \omega_1} S_{\omega_1} + \frac{\partial}{\partial \omega_2} S_{\omega_2}$ span $T_q M$ at point $q$, assuming they are linearly independent. A field $F$ on manifold assigns each $q \in M$ with a vector in $T_q M$, i.e., $F : M \to \bigcup T_q M$. Let $[t_1, t_2] \subset \mathbb{R}$ a connected interval. We denote a differentiable parametric curve $(\omega_1(t), \omega_2(t)), t \in [t_1, t_2]$ by $\gamma(t) : [t_1, t_2] \to \Omega$, and its mapped curve by $\gamma = S(\gamma(t)) \subset M$, i.e., $\gamma$ is a differentiable parametric curve that maps $\gamma : [t_1, t_2] \to M$. We write $\frac{d\gamma}{dt}$ as $\dot{\gamma}$, often interpreted as the velocity at time $t$. Using the chain rule, $\dot{\gamma}$ can be calculated as $\dot{\gamma} = S_{\omega_1} \frac{d\omega_1}{dt} + S_{\omega_2} \frac{d\omega_2}{dt} = J_{\gamma} \dot{\omega}$ where $J(\omega_1, \omega_2)$ is the Jacobian matrix with columns $(S_{\omega_1}, S_{\omega_2})$. For a vector $\dot{\omega}$ originating from a point $(\omega_1, \omega_2)$, similarly to the velocity vectors $\dot{\gamma}$, $\dot{\omega}$ is mapped to $M$ as

$$a(q) = f(\omega_1, \omega_2) \dot{\omega},$$  

(3)

Geodesically convex sets are subsets of a manifold in which any two points have the geodesics between them contained in the set. Sometimes, it is further assumed that there is a unique geodesic between two such points.

$M$ locally behaves like $\mathbb{R}^2$. See, for instance, smooth manifold as “coordinate system” defined in Theorems 5–2 of Spivak [2018]. See also local diffeomorphism for instance in Lee [2013].
where \( q = S(\omega_1, \omega_2) \). Note that \( a(q) \in T_q \). The induced inner product on \( \Omega \) is \( \langle \hat{a}, \hat{b} \rangle_\gamma = \langle f \hat{a} \rangle^T (f \hat{b}) = \hat{a}^T (f^T f) \hat{b} \). Hence, when \( M \) is parameterized using \( (\Omega, S) \), it is equipped with the metric \( g = f^T f \).

\[
\begin{align*}
g(\omega_1, \omega_2) &= \begin{pmatrix} S_{\omega_1 \omega_1} & S_{\omega_1 \omega_2} \cr S_{\omega_2 \omega_1} & S_{\omega_2 \omega_2} \end{pmatrix}.
\end{align*}
\]

\( S_{\omega_1 \omega_1}, S_{\omega_2 \omega_2} \) are assumed to be linearly independent, thus \( g \) is positive definite (and invertible). The vector magnitude can now be calculated using \( \hat{a}, \hat{b} \) as \( ||\hat{a}||_g = ||\hat{a}|| \) and \( ||\hat{b}||_g = ||\hat{b}|| \). As a consequence, the length of \( y \) is \( L(y) = \int_C ||\gamma_t||_g dt = f_c^d ||\gamma_t||_g dt \). The normal to \( y(t) \), denoted \( n(t) \), is defined as the normalized intersection of two planes: perpendicular to \( \gamma \) at \( t \) and tangent to \( M \) on \( y = y(t) \).

Let \( C = C(\hat{\Omega}) \subset M \) with \( \hat{\Omega} \subset \Omega \). The perimeter of \( C \) is the length of its boundary \( \partial C \). Let \( y^C \) be a parameterized curve mapping \( t \in [c, d] \) to the whole of \( \partial C \), then

\[
\text{per}(C) = \int_{\partial C} dl = \int_c^d ||\gamma_t^C||_g dt = L(y^C),
\]

where the first equality expresses integration on the boundary regardless of the choice of parameterization, and \( y^C = S(y^C) \). The area of \( C \) is

\[
|C| = \int_C dM = \int_C |C|, \quad \text{da}
\]

where \( da = \sqrt{|g|} da \) is often referred to as an area element, and the first equality expresses integration on the manifold, regardless of the choice of the parameterization of \( S \) with respect to \( \Omega \).

Let two functions \( f_1, f_2 \) be defined on the manifold \( M \), with their corresponding representations in \( \Omega, u_1, u_2 \), respectively, defined in a similar manner to \( u, f \) above. Throughout this work, we consider functions to lie in a Hilbert space, commonly denoted as \( \Gamma \) (see for instance Güneysu and Pallara [2015]), in which the inner product between two such functions is

\[
\int_M f_1 f_2 dM = \int_\Omega u_1 u_2 |C| da.
\]

Similarly, a gradient operator \( \nabla_g \) that satisfies \( \langle \nabla_g u, \omega \rangle_\gamma = \lim_{t_0 \to t} \frac{f_{g(u(t_0)), \omega} f_{g(u(t)), \omega}}{t_0-t}, \quad \forall \omega \in \mathcal{T}_\gamma, \quad ||\omega||_2 = ||\omega||_g = 1 \) is obtained. The divergence operator \( \nabla_g^* \), is then given by the adjoint of \( \nabla_g \), i.e., it satisfies

\[
\int \nabla_g^* \hat{F} du = \int \langle \hat{F}, \nabla_g u \rangle_\gamma da.
\]

Finally, we can define the divergence theorem on manifolds: Let \( C \subset M \) be a compact set with a smooth boundary \( \partial C \) and a boundary normal \( n^C \). Then, \( C \) may be treated as a manifold in its own right. Let \( F \) be a vector field on \( C \) (\( F \) is compactly supported), then

\[
\int_C \nabla \cdot F dM = \int_{\partial C} F^T n^C \cdot dl.
\]

We can express Equation (7) also in the parameterization domain:

\[
\int_C \nabla_g \cdot \hat{F} du = \int_{t_1}^{t_2} \langle \hat{F}, n^C \rangle_g ||\gamma^C_t||_g dt,
\]

where \( \gamma^C_t \) is defined as in Equation (5), as a smooth parametric curve along \( \partial C \), i.e., \( \gamma^C = S(\gamma^C) \) is along \( \partial C \).

The \( \mathcal{P} \)-Laplace-Beltrami is defined as

\[
\Delta_g, \mathcal{P}(u) := \nabla_g \cdot (\nabla_g u)^{\mathcal{P}-2} \nabla_g u.
\]

For \( \mathcal{P} = 2 \), \( \Delta_g, \mathcal{P}(u) \) coincides with the Laplace-Beltrami operator, yielding a diffusion process on surfaces by

\[
\frac{\partial u}{\partial t} = \Delta_g, 2(u), \quad u(0) = f.
\]

Other special cases we will discuss are \( \mathcal{P} = 3 \) and \( \mathcal{P} = 1 \).

### 3.1 Discretization

The surfaces in our experiments are discretized as a standard piecewise-linear triangle mesh. Denote \( \{v_i\}_{i=1}^n \) as the mesh vertices and \( \{t_i\}_{i=1}^m \) as the triangles (faces). Let \( f^d : \{v_i\} \to \mathbb{R} \), a common discretization of \( f \) (a function on the surface). Denote \( f^i = f^d(v^i) \) and let \( f^i \in \mathbb{R}^d \). For the implementation of discrete operators, we rely on the popular \texttt{gptoolbox} library [Jacobson et al. 2021]. A derivation of their discrete operator matrices can be found in the Jacobson and Panuzzo [2017] tutorials (chapter 2). For a brief description of their derivations, refer to Appendix F. Denote \( \tilde{G} \) as this discrete gradient operator matrix. We define

\[
\nabla f^d := \tilde{G} \tilde{f}.
\]

This gradient vector field is composed of \( \mathbb{R}^3 \) vectors, where each of the \( m \) triangles has a single vector. For convenience, the vector field is stacked to a vector in \( \mathbb{R}^{3m} \), i.e., \( [G] \in \mathbb{R}^{3m \times n} \). Let \( h \) be a vector field on triangles, stacked to a \( \mathbb{R}^{3m} \) vector (as is the gradient field \( \tilde{G} \tilde{f} \)). We can derive a divergence matrix \( [D] \) by requiring it to be Hermitian conjugate of \( [G] \) as follows:

\[
([G] \tilde{f})^T [A] \tilde{h} = \tilde{f}^T ([M][D] \tilde{h}) \tilde{f}, \quad \Rightarrow [D] = [M]^{-1} [G]^T [A].
\]

4 RE-FORMULATING TV ON MANIFOLDS AS TV ON PARAMETERIZED SURFACES

In this section, we present in detail the fundamentals of TV on parametric surfaces, re-formulated from the literature about the more abstract TV on Riemannian manifolds, see for instance Miranda Jr [2003] and Ben-Artzi and LeFloch [2007]. This bridges between complex mathematical concepts to applied computer graphics research and serves our theoretical investigations and numerical demonstrations, presented later.

We analyze functions on surfaces in the setting of Section 3. Let \( M \) be a smooth manifold given as a differentiable and invertible parametric surface \( S(\Omega), \Omega \subset \mathbb{R}^2 \) with domain variables \( \omega_1, \omega_2 \in \Omega \). The Jacobian \( f \) maps vectors from \( \Omega \) to \( M \), inducing the metric \( g \) (see Equations (2), (4)). We assume to have a function \( u : \Omega \to \mathbb{R} \) and a field \( z : \Omega \to \mathbb{R}^2 \). We assume that \( z \) is differentiable with
compact support. Note that $u$ is not necessarily continuous. We examine the following non-Euclidean TV functional,

$$NETV(u) := \sup_z \int \Omega u \nabla_g \cdot z \, da \text{ s.t. } \|z\|_g \leq 1 \forall \omega_1, \omega_2 \in \Omega. \quad (15)$$

This is a special case of TV on Riemannian manifolds, reduced to our parametric setting of two-dimensional surfaces (compatible with mesh processing). For the general case, see Ben-Artzi and LeFloch [2007] and Miranda Jr [2003].

$NETV(u)$ has two notable special cases: the Euclidean metric, which is obtained for a planar $M$, and the non-Euclidean integral formulation $\int_\Omega \|\nabla_g u\|_g \, da$, which is obtained for a differentiable function $u$. In this article, our focus is on non-Euclidean metrics and non-continuous functions (unless otherwise noted).

Usually, a Neumann boundary condition is assumed, achieving invariance to shift by a scalar,

$$NETV(u + \alpha) = NETV(u), \forall \alpha \in \mathbb{R}. \quad (16)$$

Note that if $M$ is closed, then the boundary is empty—hence, the Neumann boundary condition holds trivially. Another commonly used property is that $NETV$ is absolutely one-homogeneous, i.e.,

$$NETV(\alpha u) = |\alpha|NETV(u), \forall \alpha \in \mathbb{R}. \quad (17)$$

If a field $z$ admits the supremum of $NETV(u)$ (Equation (15)), then the field $\text{sign}(u)z$ admits $NETV(\alpha u)$. For an in-depth derivation and generalization of basic properties such as the above, see Miranda Jr [2003], Section 3.

Being absolutely one-homogeneous, we know that the subdifferential of $NETV$, as stated for instance in Burger et al. [2016], is the following set:\footnote{The deduced subdifferential needs a product space. We consider the definition of Equation (7).}

$$\partial NETV(u) = \left\{ p : NETV(v) \geq \int_\Omega p \nu \, da \forall v, NETV(u) = \int_\Omega p \, da \right\}, \quad (18)$$

where $v$ is assumed to have the same Neumann conditions as $u$. By Equations (18), (15), we have

$$\nabla_g \cdot z \in \partial NETV(u) \Rightarrow \left\{ \begin{array}{l} \text{NETV}(u) = \int_\Omega u \nabla_g \cdot z \, da. \quad (19) \\ \text{NETV}(v) \geq \int_\Omega v \nabla_g \cdot z \, da, \quad (20) \end{array} \right.$$

and the converse

$$\int_\Omega u \nabla_g \cdot z \, da = NETV(u) \Rightarrow \nabla_g \cdot z \in \partial NETV(u). \quad (21)$$

The $NETV$ minimizing flow, performed on a function $f : \Omega \rightarrow \mathbb{R}$, is defined as

$$u_t = -p(t), \quad p(t) \in \partial NETV(u(t)), \quad u(0) = f, \quad t \geq 0, \quad (22)$$

where $u_t = \frac{\partial u}{\partial t}$. Recently, Bungert and Burger [2020] proved that eigenfunctions are exposed upon decay by such flows as asymptotic solutions (just before extinction). This enables us to reveal eigenfunctions numerically, by simulating Equation (22), as done in Figures 2, 3, 4.

4.1 Indicator Functions

Indicator functions are non-continuous functions that have an important role in total variation analysis. In our non-Euclidean setting, we analyze the indicator function of a subset $C \subset M$,

$$\chi^C(q) = \begin{cases} 1 & q \subset C \\ 0 & q \subset M \setminus C \end{cases}, \quad (23)$$

for which we construct

$$\tilde{\chi}^C = \chi^C \circ S, \quad (24)$$

i.e., $\tilde{\chi}^C$ is the indicator of $C \subset \Omega$, where

$$\tilde{C} = \{ S^{-1}(q) \mid q \in C \}. \quad (25)$$

For convenience, let us define the $NETV$ of a set as the $NETV$ of its indicator function $NETV(C) := NETV(\tilde{\chi}^C)$, i.e.,

$$NETV(C) := \sup_{\tilde{\chi}^C} \int_{\tilde{C}} \nabla_g \cdot z \, da \text{ s.t. } \|z\|_g \leq 1 \forall \omega_1, \omega_2 \in \Omega. \quad (26)$$

In the following, we assume the setting in which the divergence Theorem on manifolds (10) holds, i.e.: $C$ is a connected set with a smooth boundary $\partial C$. The boundary has normals $n^C$ (perpendicular to $\partial C$ and tangent to $M$) with a corresponding $n^C \text{ s.t. } n^C = J_f C$, where $J$ is the Jacobian from Section 3. In addition, we assume a parameterized curve $\gamma_C : t \in [t_1, t_2] \rightarrow \partial C$ for which we construct $\tilde{\gamma}^C = S^{-1} \circ \gamma_C$, a parameterized curve along $\partial C$.$^6$

Let a field $z$ that is normal to the boundary of $C$ on (almost all) boundary points, and of norm less than or equal to one everywhere on $M$, i.e.,

$$z = n^C f \text{ for a.e. } \omega_1, \omega_2 \in \partial C, \quad \|z\|_g \leq 1 \forall \omega_1, \omega_2 \in \Omega, \quad (27)$$

$^6$To understand how $S$ maps boundaries from one domain to another refer to Lee [2013], Theorem 2.18.
where a.e. stands for "almost every." Then, z admits the supremum of Equation (26) for \( \text{NETV}(C) \). Such a z exists if the boundary of C is differentiable almost everywhere. This condition is satisfied by the assumption of a smooth boundary.

Last property but not least, we have that
\[
\text{NETV}(C) = \text{per}(C). 
\]  
(28)

For a proof and further details, see Section B in the Appendix.

5 THEORETICAL FINDINGS

Here, we generalize the theory from the Euclidean setting to closed non-Euclidean manifolds\(^7\) of the parametric surface setting (Section 4). Ultimately a new generalization of convexity is derived, as we prove properties of the eigenfunctions of the sub-gradient. The theory is demonstrated numerically as well.

5.1 Derivations

Definition 5.1. \( u \) is an eigenfunction of \( \text{NETV} \) if \( \exists \xi : \Omega \rightarrow \mathbb{R}^2 \) s.t.
\[
\begin{align*}
  \text{a.} & \quad \lambda u \in \partial \text{NETV}(u) \\
  \text{b.} & \quad \nabla_g \cdot \xi = \lambda u \\
\end{align*}
\]
for some \( \lambda \in \mathbb{R} \).

Definition 5.2. \( C \) is an eigenset if \( \exists \beta > 0 \) s.t.
\[
\psi := \tilde{\chi}^C - \beta \tilde{\chi}^{M\setminus C}
\]  
(29)
is an eigenfunction of \( \text{NETV} \), where \( M\setminus C \) is the complement of \( C \), and \( \tilde{\chi}^{M\setminus C} \) is defined as in Equation (24).

The existence of eigensets should be verified for each \( M \) individually. We provide an example of such verification for a torus in the Appendix. We also find eigensets for various manifolds in our numerical experiments.

Claim 1. If \( C \) is an eigenset with \( \psi \) as in definition (29) and \( \lambda \neq 0 \), then
\[
\beta = \frac{|C|}{|M\setminus C|},
\]  
(30)
where \( |C|, |M\setminus C| \) are areas of \( C, M\setminus C \).

Proof. By manifold divergence theorem (see for instance proposition 4.9 in Gallot et al. [1990]), we have for closed manifolds that \( \int_{\Omega} \nabla_g \cdot \xi \, z \, da = 0 \), since their boundary is an empty set.\(^5\) \( \psi \) is an eigenfunction, i.e., \( \exists \xi \) as in Definition 5.1, namely, \( \nabla_g \cdot \xi = \lambda \psi \). Plugging this, we have \( \int_{\Omega} \nabla_g \cdot \xi \, da = \int_{\Omega} \lambda \psi \, da = 0 \). Plugging definition (29), we get
\[
\lambda \int_{\Omega} \tilde{\chi}^C - \beta \tilde{\chi}^{M\setminus C} \, da = 0.
\]  
(31)
Thus, since \( \lambda \neq 0 \), by Equation (6),
\[
|C| - \beta |M\setminus C| = 0.
\]  
(32)

We note that in case \( C = M \), we have by Equation (31) that \( \lambda |M| = 0 \), i.e., \( \lambda = 0 \). This settles well with the following claim:

\(^5\)Note that closed manifolds induce a different boundary condition than the non-closed case. The treatment of both cases is similar—for brevity, we show the closed case only.

\(^7\)See for instance chapter 4 A.2 in Gallot et al. [2004].

Fig. 3. \( \text{NETV} \) minimizing flow on a spherical manifold \( M \). The initial function \( f \) assumes two options: an indicator function of a geodesically convex spherical cap \( C \) (lower row) or of \( M\setminus C \), which is not geodesically convex (upper row). From right to left: Values of \( q_1 \in C \) throughout the flow, Values of \( q_2 \in M\setminus C \) throughout the flow, and the flow portrayed as color throughout the flow, and their values change linearly with time \( t \) until they decay completely. Such a behavior implies that both \( f \)'s are eigenfunctions, i.e., \( C \) and \( M\setminus C \) are eigensets. Notably, \( M\setminus C \) is not geodesically convex. However, in the Euclidean case, eigensets must be convex sets. This raises the question: Is there an alternative notion of convexity on manifolds, other than geodesic convexity, which characterizes eigensets similarly to the Euclidean case? In the following, we define and prove such a notion.

Claim 2. If \( C \) is an eigenset, then
\[
\lambda |C| = \text{NETV}(C). 
\]  
(33)

Proof. Let \( \psi \) as in definition (29). Plugging the eigenfunction property \( \nabla_g \cdot \xi = \lambda \psi \) into \( \int_{\Omega} \psi \nabla_g \cdot \xi \, da \) yields
\[
\int_{\Omega} \psi \nabla_g \cdot \xi \, da = \lambda \int_{\Omega} \psi^2 \, da = \lambda (|C| + \beta^2 |M\setminus C|) = \lambda |C|(1 + \beta),
\]  
(34)
where the last equality uses Equation (30) as follows: \( |C| + \beta^2 |M\setminus C| = |C|(1 + \beta) \). However, we use Definition 5.1 again, namely, \( \nabla_g \cdot \xi = \partial \text{NETV}(\psi) \)–which we plug to Equation (19) and get
\[
\int_{\Omega} \psi \nabla_g \cdot \xi \, da = \text{NETV}(\psi) = \text{NETV}((1 + \beta) \tilde{\chi}^C - \beta) = (1 + \beta) \text{NETV}(C),
\]  
(35)
where the last equality is by Equations (17), (16), and the equality before uses \( \psi = \tilde{\chi}^C - \beta \tilde{\chi}^{M\setminus C} = (1 + \beta) \tilde{\chi}^C - \beta \). Thus, we have by Equations (34) and (35) the relation
\[
\int_{\Omega} \psi \nabla_g \cdot \xi \, da = |\lambda| |C|(1 + \beta) = (1 + \beta) \text{NETV}(C),
\]  
(36)
i.e.,
\[
|\lambda| |C| = \text{NETV}(C).
\]  
(37)

Corollary. Plugging Equation (28) to Equation (33), we have
\[
|\lambda| |C| = \text{NETV}(C) = \text{per}(C).
\]  
(38)

Definition 5.3. A set \( C \subset M \) with a nonempty \( \partial C \) is called a minimal perimeter set if every set \( D \) that contains \( C \) has a larger perimeter, i.e.,
\[
\text{per}(C) \leq \text{per}(D) \forall D \supset C.
\]  
(39)
Definition 5.4. A set \( C \subset M \) with a nonempty \( \partial C \) is called a locally minimal perimeter set if

\[
\text{per}(C) \leq \frac{|M|C}{|M\setminus D|} \text{per}(D) \quad \forall D \supseteq C. \tag{40}
\]

This definition is weaker than the previous definition, since \( \frac{|M|C}{|M\setminus D|} > 1 \). The locality of this definition can be explained as follows: The larger \( \frac{|M|C}{|M\setminus D|} \) is, the further away some points in \( D \) must be from the local neighborhood of \( C \). Notable cases:

- Any minimal perimeter set is also a locally minimal perimeter set.
- Closed surfaces have \( \text{per}(M) = 0 \), thus they do not contain minimal perimeter sets, but they do contain locally minimal perimeter sets.
- In the Euclidean case, both these definitions coincide with the set being convex.

Theorem 5.5. Any eigenset \( C \), a subset of a closed \( M \), satisfies

\[
\text{NETV}(C) \leq \frac{|M|C}{|M\setminus D|} \text{NETV}(D), \quad \forall D \supseteq C. \tag{41}
\]

Proof. By the eigenset Definition 5.1, \( \exists \psi \) as in definition (29) with a field \( \xi \) s.t. \( \nabla g \cdot \xi \in \partial \text{NETV}(\psi) \). Thus, by Equation (20), we have

\[
\int \chi^D \nabla g \cdot \xi \, da \leq \text{NETV}(\chi^D), \tag{42}
\]

where \( \chi^D \) is defined similarly to \( \chi^C \). By definition of \( \chi^D \), we have \( \int_D \chi^D(\cdot) \, da = \int_D \chi(\cdot) \, da \). Using this and the notation of Equation (26), we have

\[
\int_D \nabla g \cdot \xi \, da \leq \text{NETV}(D). \tag{43}
\]

However, we have the eigenset property \( \nabla g \cdot \xi = \lambda \psi \). Plugging this to \( \int_D \nabla g \cdot \xi \, da \), we have

\[
\int_D \nabla g \cdot \xi \, da = \lambda \int_D \psi(\cdot) \, da = \lambda (|C| - \beta (|D| - |C|)). \tag{44}
\]

Note that we integrate over \( D \), while \( \psi \) is defined for \( C, M \). The last equality uses \( C \subset D \), which translates to \( C \subset \hat{D} \) by invertibility of \( S \). Thus, by Equations (44), (45),

\[
\lambda (|C| - \beta (|D| - |C|)) \leq \text{NETV}(D). \tag{45}
\]

Plugging Equations (33), (30), we get

\[
\text{NETV}(C) \left( \frac{|C|}{|M|C} \frac{|D| - |C|}{|M\setminus D|} \right) \leq \text{NETV}(D), \tag{46}
\]

\[
\text{NETV}(C) \left( \frac{|C|}{|M|C} \frac{|M\setminus (C - (D - |C|))}{|M\setminus D|} \right) \leq \text{NETV}(D), \tag{47}
\]

using \( |M\setminus C| - (|D| - |C|) = |M| - |C| - |D| + |C| = |M\setminus D| \), we finally have

\[
\text{NETV}(C) \leq \frac{|M|C}{|M\setminus D|} \text{NETV}(D). \tag{48}
\]

Corollary. Plugging Equation (28) to Equation (48), we obtain

\[
\text{per}(C) = \text{NETV}(C) \leq \frac{|M|C}{|M\setminus D|} \text{NETV}(D) = \frac{|M|C}{|M\setminus D|} \text{per}(D) \quad \forall D \supseteq C, \tag{49}
\]

Fig. 4. Weighted indicator functions on Torus and their \( \text{NETV} \) flow. First row: Two geodesical disks. The small disk is defined in “Euclidean position,” i.e., results for Euclidean case carry over. Namely, the small disk is both geodesically convex and a (locally) minimal perimeter set. The large disk is not in Euclidean position, and Euclidean laws do not carry over: It is not geodesically convex, nor is it of locally minimal perimeter. Throughout the flow it becomes a sleeve-like set, which is a minimal perimeter set, after which it decays completely. Bottom row: Indicator function of two sleeve sets \( C_1, C_2 \), weighted by the ratio of their areas: \( f = \tilde{\chi} C_1 + \frac{|C_1|}{|C_2|} \tilde{\chi} C_2 \). The sleeve sets are eigensets, as shown in the Appendix. By Equation (38), and since \( \text{per}(C_1) = \text{per}(C_2) \), the eigenvalue ratio is expected to be the inverse of the ratio between the sets’ areas, i.e., \( \frac{|C_2|}{|C_1|} \). Thus, we can reformulate as \( f = \tilde{\chi} C_1 + \frac{|C_1|}{|C_2|} \tilde{\chi} C_2 \). If this is true, then by Equation (53), we expect the sets to completely decay at the same time—and indeed they do.

thus any eigenset \( C \), subset of a closed \( M \), is a locally minimal perimeter set.

If \( |M| = \infty \), then \( C \) is a minimal perimeter set. This happens because locally minimal perimeter subsets of surfaces with \( |M| = \infty \) automatically become minimal perimeter, by \( \frac{|M|C}{|M\setminus D|} = 1 \). Furthermore, if \( M \) is Euclidean (with \( |M| = \infty \), then this corollary shows that \( C \) is convex, which is a well-known property for Euclidean TV. Hence, the new notion of locally minimal perimeter is a generalization of set convexity, with regard to total-variation theory.

5.2 Overview: Application of the Theory

Eigenfunctions of the non-Euclidean total variation functional are stable, linearly decaying, modes of the minimizing flow. They are expected to be seen throughout the flow, especially upon convergence (Gilboa [2013], Bungert and Burger [2020]), Figure 3 demonstrates such linear decay, indicating that the decaying function is an eigenfunction. Such eigenfunctions divide the non-Euclidean manifold to subsets, called “eigensets.” For the Euclidean setting, it is well established that such eigensets must be convex, giving interpretability and understanding of the expected behavior of the total-variation minimizing flow, and its regularization properties in general (Andreu et al. [2001], Bellettini et al. [2002]). Until now, a fitting generalization was not proven for the non-Euclidean setting.

It was recently proposed [Fumero et al. 2020], based on empirical evidence, that eigensets of the non-Euclidean total variation are geodesically convex. However, Figure 3 demonstrates numerically a linearly decaying set, i.e., an eigenset, that is not geodesically convex.

In our theoretical derivations, we begin by finding the expected values of the eigensets. These enable us to find the eigenvalue
associated to the eigenfunction. The eigenvalue readily gives us understanding of the flow behavior, as it is equal to the rate of linear decay of the eigenfunction. Figure 4 demonstrates numerically that the rate of decay is as expected. Finally—we derive a necessary condition for a set to be an eigenset, which we call the “locally minimal perimeter” condition. This condition is validated to be a non-Euclidean generalization of the well-known Euclidean set convexity property. In Figure 2, we observe the locally minimal perimeter criterion on the stable modes of the flow. For the numerical flow, see Algorithm 2 in the Appendix. The influence of the locally minimal perimeter criterion can also be seen across our shape processing experiments that follow, most notably in Figures 5, 11, 12, 13, 18.

6 SPECTRAL DECOMPOSITIONS

Here, we show straightforward extensions of some of the observations done in Burger et al. [2016] to our settings. Let $X$ be a space of functions on a surface $M$ equipped with a metric $g$, as described in Equations (2), (4). Let $p : X \rightarrow X$ be a zero-homogeneous operator, i.e.,

$$ p(\alpha f) = \text{sign}(\alpha)p(f), \quad \alpha \in \mathbb{R}, f \in X, $$

with $p(0) = 0$. We examine the following flow:

$$ u_\tau = -p(u(t)), \quad u(0) = f \in X, \quad t \geq 0, $$

where $u_\tau = \frac{\partial u}{\partial t}$. We assume the flow exists and that the solution is unique. Unlike Burger et al. [2016], here, $p$ maps functions on non-Euclidean domains. Nevertheless, the time domain, denoted by $t$, is Euclidean. Note that elements in $\partial NETV$ are zero-homogeneous, thus, $NETV$ flow (subgradient descent of the energy) is a zero-homogeneous flow. We also assume the second time-derivative of $u$ exists in the distributional sense almost everywhere and define $\phi : t \rightarrow X$ as

$$ \phi(t) = t \cdot u_{tt}. $$

Let $\psi$ be an eigenfunction with respect to $p$ with a positive eigenvalue, i.e., $\exists \lambda \in (0, \Lambda < \infty) : p(\lambda \psi) = \lambda \psi$. Let $f = \psi$, then the solution of Equation (51) is

$$ u(0) = \psi \Rightarrow u(t) = \begin{cases} (1 - \lambda t)\psi & t \leq \frac{1}{\lambda} \\ 0 & t > \frac{1}{\lambda}. \end{cases} $$

This can be verified by having, for $t < \frac{1}{\lambda}$, the relation

$$ p(u(t)) = p((1 - \lambda t)\psi) = p(\lambda \psi) = -u_\tau, $$

where the second equality uses 0-homogeneity of $p$, the third equality uses the eigenfunction property, and the last equality is an evaluation of $u_\tau$ by Equation (53). For $t = \frac{1}{\lambda}$, we have a steady state, since $p(0) = 0$. By uniqueness of the solution, we are done.

We note that, since we are examining smoothing processes, $p$ in general is a positive semidefinite operator, $(f, p(f)) \geq 0, \forall f \in X$. Thus, the eigenvalues are positive. In the case of negative eigenvalues, the flow diverges (but for a finite stopping time can still have a solution).

7 SHAPE PROCESSING

We suggest three methods for nonlinear filtering of shapes in the framework of Section 6. The methods differ by the choice of the operator $p$ of the respective flow. Each method is inspired by a different flow: $M1$ by the Heat Flow, $M2$ by cMCF, and $M3$ by MCf. $M3$ uses $p \in \partial NETV$, hence theory of Section 5 applies. $M1$ and $M2$ use different related operators, and theory regarding these is left as future work. Nevertheless, in all three methods, we attain good feature control via manipulation of the spectral components. We will also demonstrate how the choice of different operators $p$ induces different qualities.

So far, we processed a function $f$ on a 2D manifold $M$ embedded in 3D, via Equations (51), (52), (53), (55). Here, we wish to process the manifold itself—and for this purpose, we will choose a function $f$ that describes $M$. In the setting of parameterized surfaces, the surface function may be the function of choice, i.e., $f = S, \,$ a vectorial function with three channels as the three coordinate functions $x_0(o_1, \omega_2), y_0(o_1, \omega_2), z_0(o_1, \omega_2)$. Note that $S$ also induces the intrinsic metric $g$. This choice is widely used for shape flows, thus it enables a comparison between our framework and the classical ones. Other representations can be used (see for example Figure 6).

Denote the evolving shape at time $t$ as $S(t)$, and denote $c(t)$ as any evolving coordinate function of $S(t)$, that is $c(t)$ may assume $x(t), y(t), z(t)$. 7.1 Modifying Flows for Nonlinear Spectral Processing

Our framework requires a zero-homogeneous flow evolving on a fixed metric, which is required to induce spectral linear decay of the eigenfunctions. Denote the metric induced by the initial shape $g_0$. It is fixed throughout the flow. Denote the evolving shape’s metric $g_t$, which changes throughout the flow, i.e., it is not fixed.

Examining Heat Flow, MCF, and cMCF, we find that none of these flows is zero-homogeneous, and Heat Flow is the only one performed on a fixed metric. Hence, adaptations of these flows are required.
Fig. 5. Upper row: M is the surface seen also in Figure 2. f is initialized as the z coordinate function, shown in color on the shape. A zero-homogeneous operator, \( \phi \text{ONETV} \), induces the spectral components—used for filtered reconstruction. Middle row depicts filtered coordinate function as color on shape. Bottom row: Having a geometrical meaning, the filtered coordinate function can replace the shape’s coordinate. Bottom row transforms middle row in this manner, followed by a simple graph cut for clearer visibility of vertex location. The LPF (low pass filter) H1 exposes an underlying geometrical structure of three planes, corresponding to three “sleeve sets.” The sleeve sets decay linearly throughout the flow, proving to be spectral components of \( \phi \text{ONETV} \). The geometrical meaning of linear decay is constant velocity. H2 further filters and shrinks the upper two planes to one, showing that the bottom plane is of lowest eigenvalue, as it is the slowest decaying plane. H3 shrinks the three underlying planes to a single one, the residual plane. H3 adds the non-planar structures to the residual plane. The naive choice of coordinate function is not always natural to the processed shape.

7.2 Naive Method: Unpaired Coordinate Spectral TV

The naive approach utilizes a modification of Heat Flow for our framework. Heat Flow processes each coordinate function independently via Equation (12), utilizing the Laplace-Beltrami on the fixed metric \( g_0 \) throughout the flow. Thus, it satisfies a fixed metric, but it is not zero-homogeneous, and a modification is required. By replacing the Laplace-Beltrami with the 1-Laplace-Beltrami of Equation (11), zero-homogeneity is achieved, which results in the operator \( -p_{\text{Naive}}(c) := \Delta_{g_0,1c} \), and a per-coordinate flow is defined by setting \( p(u(t)) = -p_{\text{Naive}}(c(t)) \) in Equation (51). Each channel evolves separately, hence the name “unpaired coordinates.” We can now perform nonlinear spectral filtering as in Equation (55), demonstrated on the meteor model in Figure 7. Note the axis squaring effect, which violates rotation invariance and also restricts the underlying structure, resulting with bad separation between structure and detail. With that said, it does provide relatively aesthetic results if one desires such squaring effect.

7.3 Method 1 (M1): Shape Spectral TV

Here, we take into account shape coordinate inter-correlations, i.e., we go from coordinate to shape processing. We apply a vectorial flow (as in VTV) on meshes, which results in the operator

\[
-p_{M1}(c) := \nabla_{g_0} \left( \frac{\nabla_{g_0} c}{\sqrt{\sum_{c=x,y,z} |\nabla_{g_0} c|^2}} \right). 
\]  

Note that the metric is fixed as \( g_0 \). We can also verify that the operator is zero-homogeneous.

Remark: Similar flows were proposed in the past. One important example is Elmoataz et al. [2008], from which we borrow the combined gradient magnitude of the denominator—designed to account for the inter-correlation of the coordinate function. With that said, the gradient and divergence operators used in Elmoataz et al. [2008] are significantly different, as they are obtained for general graph structures, while we use operators that account for the surface properties of the mesh. For instance, the graph gradient used in Elmoataz et al. [2008] is calculated per-vertex and has a non-fixed dimensionality that is equal to the number of edges connected to said vertex, while the surface mesh gradient we use is obtained per-triangle and is parallel to said triangle, i.e., it is embedded in \( \mathbb{R}^3 \).

The flow is followed by per-coordinate spectral processing as in Equations (52), (55)—thus, \( x, y, z \) inter-correlation is preserved. Compared to the naive approach, M1 preserves better the initial underlying structure, as demonstrated in Figure 7. Good multi-scale feature control is demonstrated as well in Figure 8.

7.4 Method 2 (M2): Conformalized 3-Laplace

Here, we modify cMCF [Kazhdan et al. 2012] to our framework by presenting a conformalized \( p \)-Laplace, as described below. Our flow inherits cMCF’s limb-head smoothing capabilities (Figure 9), which we then use for shape filtering. The metric of cMCF is...
minimizing flow. To obtain this is enforced as an operator on the fixed metric \(\nabla|g_0| \cdot \nabla g_0\), depends on the evolving shape’s metric \(g_t\). To achieve a fixed metric, we re-interpret \([g_t]\) as an operator on the fixed metric \(g_0\). This is valid, since the diffused shape defines both the diffused function as well as the evolving metric. This affects homogeneity, as shown below. We define the conformalized \(P\)-Laplace as,

\[
\tilde{\Delta}_{g,\mathcal{P}}(c) := \frac{|g_0|}{|g_t|} \nabla_{g_0} \cdot ((\nabla_{g_0} c)^P - 2 \nabla_{g_0} c).
\]

By Equation (4), we have that \(|g_t|\) is absolutely 4-homogeneous, hence \(\tilde{\Delta}_{g,\mathcal{P}}\) is \(\mathcal{P} = 3\) homogeneous,

\[
\tilde{\Delta}_{g,\mathcal{P}}(ac) = \frac{|ac|}{|a||c|} \nabla_{g_0} \cdot ((|a|^P \nabla_{g_0} c)^P - 2 |a| \nabla_{g_0} c) = \frac{|a|^3}{|a||c|} \tilde{\Delta}_{g,\mathcal{P}}(c).
\]

Thus, we choose \(\tilde{\Delta}_{g,3}\) as a zero-homogeneous modification of the conformalized Laplace. Once again, inter-correlations are accounted for, as in Elmoataz et al. [2008], yielding the operator

\[
-\rho_{M2}(c) := \left|\frac{|g_0|}{|g_t|}\right| \nabla_{g_0} \cdot \left(\frac{1}{\sum_{x,y,z} |\nabla c|^2} \nabla_{g_0} c\right).
\]

The flow is followed by nonlinear spectral filtering, Equation (55). Editing extremities, a capability inherited from our conformal 3-Laplace flow, is demonstrated in Figure 10, where extremities are in the form of human limbs and head.

7.5 Method 3 (M3): Directional Shape TV

Mesh TV smoothing typically preserves pointy surface points, e.g., tip of chin [Fumero et al. 2020] or ears [Elmoataz et al. 2008]. Here, we propose a method that preserves edges, e.g., muscle contour, similarly to TV processing of images. While M1 and M2 utilized modifications of Heat Flow and cMCF, M3 draws inspiration from MCF.

MCF already has a thoroughly researched fixed-metric zero homogeneous modification: the TV flow as applied to gray-scale images [Kimmel et al. 2000]. For a surface represented as \(S = (x, y, f(x, y))\), this modification entails constraining the evolved shape to be of the form \(\hat{S}(t) = (x, y, f(x, y), t)\). This is enforced by constraining each point on the surface to evolve in direction \(\hat{z}\) (perpendicular to the x, y plane). We note that unconstrained MCF would necessarily violate this form of \(\hat{S}(t)\), as it theoretically converges to a singular point.

Our third method aims to generalize the above direction-constraint to general shapes, hence the name “directional.” The x, y domain is generalized to be an over-smoothed version of the initial shape, which we denote \(\hat{S}\). Each \(p \in S\) is mapped to a \(\hat{p} \in \hat{S}\). The direction of evolution is fixed as \(\hat{d} = (0, 0, \sqrt{\alpha(S - \hat{S})})\), where \(\alpha\) is a sign indicator that ensures \(\hat{d}\) points “outwards.” Finally, the evolving initial surface is represented as \(f = \alpha|S - \hat{S}|\). Note that \(S = \hat{S} + \alpha \hat{d}\). This method is a generalization in the following sense: Consider the form \(S = (x, y, f(x, y))\), choosing \(\hat{S} = (x, y, 0)\), we have that \(\hat{d} = \hat{z}\) and \(\alpha|S - \hat{S}| = (0, 0, f(x, y))\).
Spectral Total-variation Processing of Shapes - Theory and Applications

Fig. 8. M1 filtering. Top row, left to right: All-pass reconstruction; residual; 3-bands amplification. Bottom row: each of the 3 bands isolated on top of the residual. As t grows, choosing \( H(t) \) greater or less than 1 results in amplification or attenuation of coarser details, as can be observed, e.g., in the model’s contour and clumping of hair-strands.

Fig. 9. Three upper rows: M2 conformalized \( P \)-Laplace flows. \( P = 2 \) is an unscaled version of the conformalized Mean Curvature Flow (cMCF). \( P = 1 \) is a new conformalized shape TV flow. For \( P = 3 \) the flow is zero-homogeneous. Bottom row: cMCF from Kazhdan et al. [2012].

This method belongs to a vast family of shape processing techniques, operating on the displacement field.

We advocate the choice of \( \hat{S} \) as a cMCF smoothed version of \( S \), since cMCF was shown to provide a conformal mapping from \( S \) to \( \hat{S} \). By construction, the metric is fixed and inter-correlations are accounted for. The proposed zero-homogeneous operator (acting on a scalar-valued function \( u \)) is

\[
-\rho_{M1}(u) := \nabla g_0 \cdot \frac{\nabla g_0 u}{|\nabla g_0 u|},
\]

where \( u(t) \) is the evolution of \( f \) at time \( t \), which results in

\[
\frac{\partial S}{\partial t} = \nabla g_0 \cdot \frac{\nabla g_0 u}{|\nabla g_0 u|} \hat{d},
\]

satisfying the imposed directionality. Finally, \( f \) is filtered as in Equation (55), and a filtered shape is obtained by \( \hat{S} + f_{\text{filtered}} \hat{d} \). Being closely related to spectral TV on images, this method preserves detail well, as demonstrated in Figure 11.

Fig. 10. Shape exaggeration by M2 spectral filtering. The filter inherits its properties from the conformalized 3-Laplace flow (Figure 9), which translates to interesting limb-head editing. Isometry robustness is demonstrated as well.

Though inspired by MCF, the proposed flow is substantially different.

8 SHAPE DEFORMATION

Here, we show a novel application. For the first time, we perform total-variation for the shape deformation task. We will see that this induces a piecewise-constant deformation field, where the deformation concentrates on small-perimeter boundaries, showcasing the eigenset properties derived in Section 5.

The shape deformation problem involves finding a plausible transformation of a given shape while accommodating constraints. The constraints are specified by the user as points, or regions of the shape, which are to be moved away from their original location. The deformation process should ensure that the resulting shape maintains its structural integrity while sufficing the constraints.

To this end, we propose a constrained minimization of the total-variation of the displacement field, defined as

\[
d' := S' - S,
\]

where \( S \) is the coordinate function of the original given shape, and \( S' \) is the resulting shape. The minimization process is provided below as Algorithm 1. The algorithm uses a minimization process proposed by Bronstein et al. [2016] for minimizing L1 norms on manifolds. A slight difference is that we use a vectorial version of Bronstein et al. [2016]. The minimizing operator of choice is \( \rho_{M1} \) of Equation (56). Minimization is performed while enforcing the deformation constraints.

As an initial solution to the process, we use the gradient-based linear deformation proposed in Botsch and Sorkine [2007]. We use a matrix version of \( \rho_{M1} \), denoted \( \rho_{M1}^{\text{matrix}} \), which is updated at each iteration. Further details are available in Appendix E.

As illustrated in Figure 12, the displacement field \( d' \) gradually becomes piecewise-constant during the shape deformation process. Interestingly, the deformation tends to concentrate on boundaries with small perimeters, suggesting a potential connection to eigenfunctions. We validate this hypothesis numerically in Figure 13 by analyzing \( \|d'\|_g \), the magnitude of the field. See Figure 14 for method comparison. For completeness, we added additional results in the Appendix, Figure 20.
Fig. 11. Upper row: Low-pass filtering applied for smoothing. All our methods can utilize rough time discretization for runtime efficiency at the expense of reconstruction error. In this example, M3 is used and completed in approximately 30 seconds. In contrast, the Laplace-Beltrami method involves SVD for 2,000 eigenvectors of a $[17 \times 10^3 \times 17 \times 10^3]$ sparse matrix, requiring about 5 hours (2.5 orders of magnitude slower). Nonetheless, more efficient linear methods exist, such as in Cignoni et al. [2005], which take the same amount of time. Bottom row: Smooth caricaturization via bandpass exaggeration coupled with smoothing. We compare the exaggeration capabilities of M3 with those in Cignoni et al. [2005], which also exaggerates shapes using the deformation from an over-smoothed underlying shape. Bottom left (zoom-in): Our approach effectively smoothes smaller details, such as the scales of the armadillo, while maintaining larger structures like the knees. Our theoretical findings correlate “resistance to smoothing” of geometrical structures with generalized convexity and the ratio of perimeter-to-area. The knees, being more convex-like with a lower perimeter-to-area ratio than the scales, indeed demonstrate greater resistance to smoothing. This results in superior separation of detail compared to both Laplace-Beltrami smoothing and the shape exaggeration method in Cignoni et al. [2005]. For additional reference, exaggeration without smoothing is presented in the Appendix, Figure 19, where we also demonstrate consistency with the caricaturization principles discussed in Sela et al. [2015].

9 DISCUSSION

9.1 Non-differentiable Shapes: A Limitation of Our Theory

Our theoretical findings rely on the notion of “good” metric spaces [Miranda Jr 2003]. In the context of our shape processing applications, these metrics are defined by the specific shapes being processed. However, within the domain of computer graphics, the shapes encountered can often exhibit high non-differentiability, which may even be enhanced by discretization. Consequently, our assumptions may not be suitable in such cases. While works concentrating on such considerations were conducted for the Laplacian operator (see for instance Wardetzky et al. [2007] and Sharp and Crane [2020]), this remains for future work regarding the operators used in this article.

ALGORITHM 1: Shape Deformation TV

| Require: $V_{initial}$ | Initial mesh vertices, $n \times 3$ |
|------------------------|----------------------------------|
| Require: $F$           | Mesh faces                        |
| Require: $b$           | Boundary indices                  |
| Require: $\kappa$      | Boundary constraints              |
| Require: $w \leftarrow \infty$ | A high constraint weight |
| Require: tolerance     | Convergence criterion              |

1: $V \leftarrow V_{initial}$  // Current vertex positions

2: while true do

3: $V' \leftarrow V$  // Cache current positions

4: $[G] \leftarrow \text{gradient matrix}(V, F)$  // See (13)

5: $[D] \leftarrow \text{divergence matrix}(V, F)$  // See (14)

6: $[P] \leftarrow [D] \frac{1}{\| [G](V - V_{initial}) ]^2 + \varepsilon}$  // Combined magnitude denominator

7: $V \leftarrow \text{minimize}(\| [P] (V - V_{initial}) \|^2 + w \cdot \| V(b, \cdot) - \kappa \|^2)$

8: if $\| V - V' \| < \text{tolerance}$ then

9: Break  // Convergence check

10: end if

11: end while

12: return $V$

9.2 Flow-induced Spectral Representation

Considerations

Our filtering framework is flow-based, where spectral representations are manifested as linear decaying components. Shape analysis is performed along the time domain. In contrast, the Laplace-Beltrami eigenfunctions are acquired by solving an eigenvalue problem on the non-Euclidean domain. One advantage of the flow-based framework is computational: The numerical simulation of a shape flow is often computationally cheaper than a numerical solution to an eigenvalue problem performed on the shape domain (e.g., solving the SVD of a discrete Laplace-Beltrami operator), as demonstrated in Figure 11. This results in better filtering complexity. This is opposed to the Euclidean case, where eigenvalue decomposition of the Laplacian is not needed, since it is fixed and known (Fourier basis). Simulating a flow requires discrete timesteps, resulting with a tradeoff between computational complexity and simulation accuracy, controlled by the timestep size. This is true for Euclidean and non-Euclidean settings alike. To measure the accuracy of the flow, we can test the reconstruction error of an all pass filter, i.e., $H(t) = 1 \forall t$, as demonstrated in Figure 15. Remark: Our semi-implicit implementation of the flows exhibits stability for large time-steps, even when they result in inaccuracies. This is attributed to the time-steps of both the filtering flows and the deformation flows being approximated as solutions to stable L2 minimization problems (similar stability may be found for instance in Kazhdan et al. [2012]). Additionally, the best practice we found regarding the $\varepsilon$ parameter, which ensures a strictly positive denominator (see Algorithms 1, 2) is to find the minimal value that still enables a stable flow. Finally, the $w$ parameter, which is the weight of the constraints in our Shape Deformation Algorithm 1, has to be large enough for the constraints to be accommodated.
Another consideration arises when comparing two representations, \( \phi_1(t), \phi_2(t) \). The usual \( L_2 \) distance may not be compatible on time domains and even moreso on discretized time domains. This is since \( L_2 \) does not express well the difference between small and large shifts in time. For example, consider an eigenfunction—\( \phi(t) = \delta(t - \frac{1}{2}) \), where \( \lambda \) is the eigenvalue. Thus, the \( L_2 \) measure will not be able to discriminate between eigenfunctions with similar eigenvalues to eigenfunctions with largely different eigenvalues. Thus, appropriate distance measure must differentiate small from large time perturbations. One such measure is the earth-movers’ distance. An example using this distance on \( \phi \) is portrayed in Figure 17, where co-segmentation takes place.

9.3 Spectral Image TV and Shape Spectral TV Relation

Our shape spectral TV methods require a zero-homogeneous flow performed on a fixed metric. Considering the Beltrami and TV flow equivalence presented in Kimmel et al. [1998] (see a brief reminder of this equivalence in the Appendix, Section G), we have that applying spectral TV to images is a form of shape spectral TV. Let us have a closer look at this statement: To transition from MCF to the equivalent TV flow, the flow was rephrased on a fixed metric (the Euclidean pixel grid), which was absorbed as a nonlinearity of the operator, making it zero-homogeneous. Thus, all requirements of the zero-homogeneous spectral framework were met.

With that said, this framework is more restrictive than our general framework in three ways: The operator suggested is one of a kind, the shape has to be parameterized as an image function, and the fixed metric is a Euclidean domain (the pixel-grid).

In TV, the latter constraint is not required. As shown in Biton and Gilboa [2022], \( ATV \) can be re-interpreted using a generalization of the gradient, \( \nabla A f = A(x) \nabla f \). This gradient generalization can be obtained by considering the pixel grid as the \( \Omega \) domain in Equation (2), and \( A \) as the metric \( g \) from Equation (4), induced by some unspecified \( M \). While other aspects of \( ATV \) do not coincide with the differential geometry framework we use here, it is certainly related to our work. Interestingly, in \( ATV \) the importance of parameterization domain is greater than in our framework, as the signal lies in \( \Omega \), and gradients are on \( \Omega \) as well, mapped from an equivalent TV flow, the flow was rephrased on a fixed metric (the Euclidean pixel grid), which was absorbed as a nonlinearity of the operator, making it zero-homogeneous. Thus, all requirements of the zero-homogeneous spectral framework were met.

With that said, this framework is more restrictive than our general framework in three ways: The operator suggested is one of a kind, the shape has to be parameterized as an image function, and the fixed metric is a Euclidean domain (the pixel-grid).

In TV, the latter constraint is not required. As shown in Biton and Gilboa [2022], \( ATV \) can be re-interpreted using a generalization of the gradient, \( \nabla A f = A(x) \nabla f \). This gradient generalization can be obtained by considering the pixel grid as the \( \Omega \) domain in Equation (2), and \( A \) as the metric \( g \) from Equation (4), induced by some unspecified \( M \). While other aspects of \( ATV \) do not coincide with the differential geometry framework we use here, it is certainly related to our work. Interestingly, in \( ATV \) the importance of parameterization domain is greater than in our framework, as the signal lies in \( \Omega \), and gradients are on \( \Omega \) as well, mapped from an equivalent TV flow, the flow was rephrased on a fixed metric (the Euclidean pixel grid), which was absorbed as a nonlinearity of the operator, making it zero-homogeneous. Thus, all requirements of the zero-homogeneous spectral framework were met.

With that said, this framework is more restrictive than our general framework in three ways: The operator suggested is one of a kind, the shape has to be parameterized as an image function, and the fixed metric is a Euclidean domain (the pixel-grid).

In TV, the latter constraint is not required. As shown in Biton and Gilboa [2022], \( ATV \) can be re-interpreted using a generalization of the gradient, \( \nabla A f = A(x) \nabla f \). This gradient generalization can be obtained by considering the pixel grid as the \( \Omega \) domain in Equation (2), and \( A \) as the metric \( g \) from Equation (4), induced by some unspecified \( M \). While other aspects of \( ATV \) do not coincide with the differential geometry framework we use here, it is certainly related to our work. Interestingly, in \( ATV \) the importance of parameterization domain is greater than in our framework, as the signal lies in \( \Omega \), and gradients are on \( \Omega \) as well, mapped from an
unspecified non-Euclidean domain. In contrast, our signal lies on an explicit manifold \( M \).

\( M_3 \) generalizes both spectral \( TV \) and \( ATV \) in the following sense: Considering the form \( S = (x, y, f(x, y)) \), choosing \( S = (x, y, 0) \), we have that \( \hat{d} = \hat{z} \), and \( \alpha |S - \hat{S}| = (0, 0, f(x, y)) \). Now consider two options: Option 1: \( \hat{S} \) induces \( g \), resulting with a Euclidean flow of an image function. Option 2: \( S \) induces \( g \), resulting with a non-Euclidean flow of an image function on an adapted metric. This is a form of \( ATV \), but with specified non-Euclidean surface domain. See example in Figure 16.

9.4 Future Ideas

Our flow-based spectral framework can easily be adapted to a wide collection of operators that assume the required homogeneity and a fixed metric. Inevitably, neural-networks come to mind, where homogeneity can be taken care of using normalization layers.

Our new notion of non-Euclidean convexity, the locally minimal perimeter, might have an appropriate generalization to graphs—which are also non-Euclidean. This probably involves extending our theory from parametric surfaces to Riemannian manifolds of general dimensions.

The representations we use for filtering may be used for other tasks, such as classification and segmentation—see preliminary result in Figure 17.

Additional key aspects from the Euclidean case may be generalized to our parametric surface setting, e.g., curvature bounds of eigensets, and analysis of vectorial functions.

10 SUMMARY

We presented new nonlinear spectral theoretical analysis for surfaces by generalizing nonlinear spectral theory of image processing. Based on our analysis, we proposed a general methodology for shape analysis and processing via nonlinear spectral filtering.

A key finding is our introduction of locally minimal perimeter sets, a novel generalization of cone sets to manifolds. It is derived by generalizing properties of \( NETV \) eigenfunctions. Our analysis is supported by numerical examples of minimizing flows, where numerical validation of eigenfunctions is performed by examining the decay near extinction, following the theory of Bungert and Burger [2020].

For shape nonlinear filtering, our methods extract spectral representations from smoothing flows that satisfy two requirements: zero-homogeneity and a fixed metric. We choose to process the shape in its embedding space, providing unmediated nonlinear spectral representations, yielding good feature control. To showcase the general concept, three methods are proposed, where all three are based on the same mechanism, described in Equations (52), (55). Each method holds clear distinct properties induced by its flow, allowing various shape manipulations via spectral filtering. While possessing visibly distinct properties, all three methods demonstrate good smoothing and detail enhancement capabilities. Robustness to pose variations is demonstrated as well. With respect to processing time, we note that these methods are fairly fast, as they do not require solving an eigen-problem explicitly. Additionally, we present a Total-Variation approach for addressing the shape deformation problem. Our experiments show that the deformation using our method is concentrated on plausible segment boundaries. Moreover, we have shown for several
Fig. 17. Representations that enable good feature separating filters are informative, hence may play a role in other tasks. Here, we demonstrate an initial attempt at using $M_2$'s spectral representation for co-segmentation of the Michael models in various poses: Each point (vertex) on each pose (mesh) is assigned the representation $\phi(t)|_{q \in M}$. For co-segmentation, we partitioned representation space using all representations from all poses, yielding segments that are shared across poses. Space partitioning was done via the k-means algorithm, using earth-movers’ distance instead of Euclidean distance. Earth-movers’ is a more appropriate measure, since it is better at quantifying perturbations along the $t$ dimension.

numerical cases that these boundaries relate to our theoretical findings.9

APPENDICES

A COMPLEMENTARY EXPERIMENTS

Here, we add experiments for completeness. Figure 18 bridges a gap between Figures 2 and 5. Figure 19 shows shape exaggeration similar to Figure 11 on additional poses while not using smoothing. Figure 20 extends the method comparison performed for the Snail model in Figure 14 to three other models. Figure 21 tests the $w$ parameter used in Algorithm 1. Figure 22 tests hyper-parameters of $M_1$’s flow.

B SOME PROOFS REGARDING SECTION 4.1

Claim. Let a field $z$ that is normal to the boundary of $C$ on (almost all) boundary points and of norm less than or equal to one everywhere on $M$, i.e.,

$z = \tilde{n}^C$ for a.e. $\omega_1, \omega_2 \in \partial \tilde{C}$, $||z||_g \leq 1 \forall \omega_1, \omega_2 \in \Omega$, (61)

where a.e. stands for “almost every.” Then, $z$ admits the supremum of Equation (26) for $\text{NETV}(C)$.10

Proof. Since $\tilde{\chi}^C$ is an indicator of $\tilde{C}$, we have

$\int_{\Omega} \nabla g \cdot z \tilde{\chi}^C \, da = \int_{\tilde{C}} \nabla g \cdot z \, da$. (62)

Using manifold divergence Theorem, as stated in Equation (10), we have

$\int_{\Omega} \nabla g \cdot z \tilde{\chi}^C \, da = \int_{\Omega} \langle z, \tilde{n}^C \rangle_g ||\tilde{\gamma}^C||_g \, dt$, (63)

where $\tilde{\gamma}^C = \frac{d\tilde{\gamma}^C}{dt}$. For convenience, let us reformat this as

$\int_{\Omega} \nabla g \cdot z \tilde{\chi}^C \, da = \int_{\Omega} ||z||_g \cos \theta ||\tilde{\gamma}^C||_g \, dt$, (64)

where $\theta = \angle_g (z, \tilde{n}^C) = \cos^{-1} (\frac{\langle \tilde{n}^C, z \rangle_g}{||\tilde{n}^C||_g ||z||_g})$. Maximization under the constraint $||z||_g \leq 1$ is achieved for $\theta = 0$, and $||z||_g = 1$ almost everywhere on the boundary, i.e.,

$z = \tilde{n}^C$ for a.e. $\omega_1, \omega_2 \in \partial \tilde{C} \Rightarrow \text{NETV}(C) = \int_{\Omega} \nabla g \cdot z \tilde{\chi}^C \, da$. (65)

\[\square\]
Fig. 19. Band-Pass exaggeration via $M3$ for the armadillo in various poses, where all poses experience the same filter. The band-pass exaggeration magnitude is identical to Figure 11. Unlike Figure 11, there is no smoothing of the high frequencies. The consistency across poses shows that our method can be considered as a caricaturization method by the properties posed in Sela et al. [2015].

Fig. 20. The same experiment of Figure 14 for the other shapes shown in Figure 12. As before, the shape and deformation constraints are tested for the three methods. The same properties observed in Figure 14 can be seen here as well: The deformation field is piece-wise-constant, and the deformation is concentrated on boundaries of plausible shape segments.

Remark: The “almost all” condition allows robustness to a zero-measure subset of $\partial C$ in which boundary normals are not defined (namely, points of non-differentiable $\partial C$). In such a case, $\vec{n}^C$ may be extended to satisfy

$$\begin{cases}
J\vec{n}^C = n^C \forall \omega_1, \omega_2 \in \partial C & \text{if } n^C \text{ exists} \\
||\vec{n}^C||_{\mathbb{I}} \leq 1 & \text{otherwise},
\end{cases}$$

while keeping the proof intact.

CLAIM.

$$\text{NETV}(C) = \text{per}(C).$$

(67)

Fig. 21. This figure evaluates the influence of the $w$ parameter, which is the weight of the constraints (indicated by red dots) in our Shape Deformation Algorithm (Algorithm 1). Notably, from $w = 0$ to $w = 0.6$, we have translation without noticeable deformation. For a short transition range, $w \in [0.6, 0.7]$, we observe a gradual increase in deformation. For the very wide range of $w \in [1, 3 \times 10^8]$, the deformation constraints are completely satisfied, indicating good robustness. At $w = 4 \times 10^8$ and beyond, numerical instabilities become apparent.

Fig. 22. Here, we test single timesteps of $M1$ flow for its two hyperparameters: step size ($\Delta t$) and denominator minimal value ($\epsilon$). In each subplot, we perform a single timestep starting from the initial armadillo shape. Note that $\epsilon$ can be set to be sufficiently small while maintaining stability. As expected of our semi-implicit scheme, we observe stability even under extreme values, such as $\Delta t = 10^{10}$. Such a large timestep results in a “single-step numerical flow,” for which excessive smoothing of the shape is obtained in just one iteration, making it unsuitable for detail separation and filtering. It is important to note that such extreme timestep values, which result in inaccurate flows, are strictly for the purpose of testing numerical stability and are not recommended for practical use. For instance, the reconstruction analysis in Figure 15 shows that the threshold for accurate $M1$ flow on the armadillo should be below 0.3.

This claim can be found for instance in Fumero et al. [2020]. Let us re-prove it in our setting:

PROOF. By Equation (63), we have

$$\text{NETV}(C) = \sup_{\omega} \int_{t_1}^{t_2} \langle z, \vec{n}^C \rangle_{\gamma^I} ||\vec{n}^C||_{\mathbb{I}} dt \text{ s.t. } ||z||_{\mathbb{I}} \leq 1 \forall \omega_1, \omega_2 \in \Omega.$$  

(68)
By Equation (65), we can obtain a supremum by assigning \( z = nC \cup \partial C, \omega_1, \omega_2 \in \partial C \) and have

\[
NETV(C) = \int_{t_1}^{t_2} ||nC||^2 |g| ||g|| dt = \int_{t_1}^{t_2} ||\nabla nC|| |g| dt = \text{per}(C),
\]

where the second equality uses \( ||nC|| |g| = 1 \forall \omega_1, \omega_2 \), and the last equality comes from Definition (5).

C SLEEVE SETS AS EIGENSETS ON THE TORUS

Here, we show that sleeve sets of the torus are eigensets, considering \( M \) to be a torus with big and small radii \( R, r \).

C.1 Torus Preliminaries

We choose its parametric formulation \( S \) as follows: \( S(\omega_1, \omega_2) = ((R + r \cos(\omega_1)) \cos(\omega_2), (R + r \cos(\omega_1)) \sin(\omega_2), r \sin(\omega_1)) \), where \( \omega_1, \omega_2 \in \Omega \) and \( \Omega = [-\pi, \pi] \times [-\pi, \pi] \), inducing a metric \( g(\omega_1, \omega_2) = ((R + r \cos(\omega_1))^2, 0, 0, r^2) \), resulting with

\[
\sqrt{|g|} = (R + r \cos(\omega_1)) r.
\]

Let \( z \) be a field on the torus, then its squared norm function is

\[
||z||_g = \sqrt{(R + r \cos(\omega_1))^2 z[1] + r^2 z[2]},
\]

where \( z[1], z[2] \) are components of the field. Remembering the divergence formula

\[
\nabla_g \cdot F = \frac{1}{\sqrt{|g|}} \nabla_{\omega_1, \omega_2} \cdot (\sqrt{|g|} F),
\]

(see for instance Do Carmo [2016]), we have for the torus:

\[
\nabla_g \cdot z = \frac{1}{(R + r \cos(\omega_1))} \nabla_{\omega_1, \omega_2} \cdot \left( (R + r \cos(\omega_1)) \frac{f z}{\sqrt{|g|}} \right)
\]

and

\[
\nabla_g \cdot z = \frac{\partial}{\partial \omega_1} z[1] + \frac{-r \sin(\omega_1)}{R + r \cos(\omega_1)} z[1] + \frac{\partial}{\partial \omega_2} z[2].
\]

The sleeve set of angle-length \( l \), and center at \( \omega_2 = c_0 \) is denoted in parameterization domain as \( \hat{C} \) and defined as follows:

\[
\hat{C} = \left\{ \omega_1, \omega_2 : |\omega_2 - c_0| \leq \frac{l}{2} \right\}.
\]

For convenience, \textit{w.l.o.g.}, we consider the sleeve set to have center at \( \omega_2 = 0 \), i.e.,

\[
\hat{C} = \left\{ \omega_1, \omega_2 : |\omega_2| \leq \frac{l}{2} \right\},
\]

resulting with

\[
\partial \hat{C} = \left\{ \omega_1, \omega_2 : |\omega_2| = \frac{l}{2} \right\}.
\]

C.2 Finding a Field of Required Properties

The first property we are looking for is orthogonality to the boundary, on all boundary points. Since \( g \) is diagonal, we have that \( S \) preserves angles, hence, orthogonality may be tested in parameterization domain. Furthermore, we need the field to be of unit norm on the boundary. By Equation (77), we have that \( z \) is orthogonal to the boundary and of unit norm on the boundary if \( z(\omega_1, \omega_2 = \frac{1}{2})[1] = \frac{1}{R + r \cos(\omega_1)} z(\omega_1, \omega_2 = \frac{1}{2})[1] = -\frac{1}{R + r \cos(\omega_1)} \), and \( z(\omega_1, |\omega_2| = \frac{l}{2})[2] = 0 \forall \omega_1 \).

Let \( \Theta(\omega_2) = \left\{ \begin{array}{ll} \omega_2 - \frac{1}{2} & \omega_2 \in [0, \pi) \\ \omega_2 + \frac{1}{2} & \omega_2 \in (-\pi, 0) \end{array} \right\} \). The choice

\[
z = \left( \frac{\frac{1}{R + r \cos(\omega_1)}}{R + r \cos(\omega_1)}, a \Theta(\omega_2) \right)^T,
\]

satisfies above properties. Another required property is a constant divergence inside \( C \). Let us show that this choice satisfies that as well, except for a zero-measure set of points: Plugging to Equation (74), we have

\[
\nabla_g \cdot z = -\frac{r \sin(\omega_1)}{R + r \cos(\omega_1)} \frac{\frac{1}{R + r \cos(\omega_1)}}{R + r \cos(\omega_1)} z[1] + a \frac{\partial \Theta(\omega_2)}{\partial \omega_2},
\]

where \( \frac{\partial \Theta(\omega_2)}{\partial \omega_2} = 1 \forall \omega_2 \neq 0, \pi, \) thus

\[
\nabla_g \cdot z = a \forall \omega_2 \neq 0, \pi.
\]

C.3 Demonstrating Eigensets

To prove an eigenset, we need to construct a field \( \xi \), that satisfies the eigenfunction properties (5.1) for a \( \psi \) as in definition (29). In the current case \( \psi \) is defined for a sleeve set \( C \) on a torus.

First, we note, that the appropriate field \( z \) of Equation (78) can be similarly defined for the complement set—which is a sleeve set as well, but with its center at \( \omega_2 = \pi \). It turns out that the fields for center at either \( \omega_2 = \pi \), or at \( \omega_2 = 0 \), are of the same form, up to a sign factor. Thus, we define \( \xi \) to admit Equation (78) for \( \hat{C} \) inside \( \hat{C} \), and the minus of Equation (78) for \( \Omega \setminus \hat{C} \), i.e.,

\[
\xi = \left\{ \begin{array}{ll} \frac{1}{R + r \cos(\omega_1)}, a \Theta(\omega_2) \right\}^T \omega_1, \omega_2 \in \hat{C} \\ \frac{1}{R + r \cos(\omega_1)}, a \Theta(\omega_2) \right\}^T \omega_1, \omega_2 \in \Omega \setminus \hat{C} \end{array} \right\}
\]

which has \( \nabla_g \cdot \xi = \left\{ \begin{array}{ll} a_1 \omega_1, \omega_2 \in \hat{C} \\ a_2 \omega_1, \omega_2 \in \Omega \setminus \hat{C} \end{array} \right\} \) and is unit-orthogonal to the boundary, i.e., \( z = nC \cup \partial C, \omega_2 \in \partial C \). If we set \( a_1 = 1, a_2 = \beta \), where \( \beta \) is as in Equation (30), then indeed we have the eigenset

\[\psi = \lambda \nabla_g \cdot \xi.\]

By (5.1) it is left to show that \( \nabla_g \cdot \xi \in \partial NETV(\psi) \). To show this, it is sufficient, by Equation (21), to show that \( \int_{\Omega} \nabla_g \cdot \xi \psi da = NETV(\psi) \). Let us begin: By one-homogeneity of the \( NETV \), we have that \( NETV(\psi) = (1 + \beta) NETV(C) \).

Since \( \xi = nC \cup \partial C \cup \partial C, \omega_2 \neq 0, \pi \), we have by Equation (65) that

\[\text{NETV}(\mathcal{C}) = \int_{\Omega} \nabla_g \cdot \xi \psi da.\]

Thus, by the definition of \( \psi \) (29), we have

\[\int_{\Omega} \nabla_g \cdot \xi \psi da = \int_{\Omega} \nabla_g \cdot \xi (nC - \beta \mathcal{M}(C)) da = \int_{\Omega} \nabla_g \cdot \xi (\mathcal{C} - \beta (1 - \mathcal{C})) da = (1 + \beta) \int_{\Omega} \nabla_g \cdot \xi (\mathcal{C} da - \beta \int_{\partial \Omega} nC da = (1 + \beta) \int_{\partial \Omega} \nabla_g \cdot \xi (\mathcal{C} da - \beta \int_{\partial \Omega} nC \xi da = (1 + \beta) \int_{\partial \Omega} \nabla_g \cdot \xi \mathcal{C} da,\]

where the last cancellation uses the Neumann boundary condition assumption. Thus, we have \( \int_{\Omega} \nabla_g \cdot \xi \psi da = (1 + \beta) \int_{\Omega} \nabla_g \cdot \xi \mathcal{C} \)

ACM Trans. Graph., Vol. 43, No. 2, Article 22. Publication date: February 2024.
where we use $\rho_{M_1}$ of Equation (56) as the sub-differential. See Figures 14, 12, 20 for the results.

F OPERATOR DISCRETIZATION SUMMARY

Let us briefly describe the derivation of the discrete operator matrices suffered in Jacobson and Panozzo [2017] tutorials (chapter 2). Reminder: A surface in our experiments is discretized as a standard piecewise-linear triangle mesh. Denote $\{v_i\}_{i=1}^n$ as the mesh vertices and $\{t_i\}_{i=1}^m$ as the triangles (faces). Let $f^d : \{v_i\} \rightarrow \mathbb{R}$, a common discretization of $f$ (a function on the surface). Denote $f_i = f^d(v_i)$ and let $\tilde{f}(i) = f_i$.

Let us approximate $f$ as an extrapolation of $f^d$ to all mesh points (i.e., edges and triangle points) s.t. the approximation is a continuous triangle-linear function. A triangle-linear function is a piecewise-linear function on a mesh that is linear on each of the mesh triangles. To this end, let $\phi_i(x)$ be the tent, or hat function at vertex $i$, i.e., it is the unique triangle-linear and continuous function that is equal to 1 on vertex $i$ and 0 on the other vertices. Let us extrapolate $f^d$ via the tent functions

$$f \approx \sum_{i=1}^n \phi_i(x)f_i.$$ (82)

Indeed, this approximation is a continuous triangle-linear extrapolation of $f^d$. By triangle-linearity of $\phi_i(x)$, we have that $\nabla \phi_i(x)$ is a triangle-constant vector field, i.e., it is constant within each triangle. By linearity, the gradient of the above approximation is $\sum_{i=1}^n \nabla \phi_i(x)f_i$, i.e., it is a linear combination of $\nabla \phi_i(x)$, making it triangle-constant as well. Hence, it can be expressed as a matrix multiplication taking $\tilde{f}$ values to triangle values. Denote $[G]$ as this discrete gradient operator matrix. We define

$$\nabla f^d := [G] \tilde{f}. $$ (83)

This gradient vector field is composed of $\mathbb{R}^3$ vectors, where each of the $m$ triangles has a single vector. For convenience, the vector field is stacked to a vector in $\mathbb{R}^{3m}$, i.e., $[G] \in \mathbb{R}^{3m \times n}$.

Let $h$ be a vector field on triangles, stacked to a $\mathbb{R}^{3m}$ vector (as is the gradient field $G \tilde{f}$). We can derive a divergence matrix $[D]$ by requiring it to be Hermitian conjugate of $[G]$ as follows:

$$([G]^T[A]h = [G]^T[M][Dh]\nabla \tilde{f}, \tilde{h} \Rightarrow [D] = [M]^{-1}[G]^T[A],$$ (84)

where $[M], [A]$ are diagonal matrices that have area elements as their diagonal and are constructed for functions on vertices and fields on triangles, respectively. $[A] \in \mathbb{R}^{3m \times 3m}$ has the $m$ triangle areas as its diagonal (replicated 3 times) and $[M] \in \mathbb{R}^{n \times n}$ has effective vertex areas as its diagonal, calculated as a barycentric mass matrix. For the formula and derivation of $[M]$, we refer again to the tutorials provided for Jacobson and Panozzo [2017]. This $[D]$ operator matrix is used as well in Jacobson et al. [2021], where they use the slightly different divergence convention $[D] : = [G]^T[A]$, which is often multiplied by $[M]^{-1}$ to attain $[D] = [M]^{-1} [G]^T[A]$. Note that $[D], [\bar{D}] \in \mathbb{R}^{n \times 3m}$, i.e., the divergence matrix maps triangle vector fields to vertex functions.

G SHAPE FLOWS AND THEIR EQUIVALENCE SETS

This section serves as a reminder of observations on equivalent flows, namely, the equivalence presented in Kimmel et al. [1998].
Consider the flow equation,
\[
\frac{\partial f}{\partial t} = p(f(t)),
\]
where \( p \) is some operator. E.g., choosing \( p = \Delta_{g,2} \), we obtain Equation (12). Suppose that \( M \) is a manifold of some shape, and we would like to process this shape via a flow. To do so, we can initialize the flow with the shape's coordinate function, i.e., \( f(0) = S \).

Doing so with Equation (12) is a classical shape smoothing flow. During the shape flow, at a given time \( t \), we have \( p(f(t)) : M \to \mathbb{R}^3 \), a field that describes the velocity of the evolving shape's points. Recalling the normal
\[
N = \frac{S_{\omega_2} \times S_{\omega_3}}{|S_{\omega_2} \times S_{\omega_3}|},
\]
a vector \( \hat{V} \in p(f(t)) \) may be decomposed to its normal and tangential component as follows:
\[
\hat{V}_N = (\hat{V}, N)N, \quad \hat{V}_T = \hat{V} - \hat{V}_N,
\]
yielding
\[
\frac{\partial N}{\partial t} = (p(f)N + p(f)\tau). 
\]

The normal component accounts for the change of the shape in time, and the tangential movement is merely a change of the shape's parameterization in time. Thus two shape flows are considered equal if their normal components are equal, and an equivalence set of shape flows is defined as:
\[
\left\{ \frac{\partial f}{\partial t} = q(f) : (q(f), N) = (p(f), N) \right\}.
\]

We note that parameterization constraints may cause differences between two flows from the same equivalence set.

G.1 Beltrami - TV Flow Equivalence

In Kimmel et al. [1998], an equivalence between MCF and the TV-flow was shown: On one hand, they define the Beltrami flow \( \frac{\partial f}{\partial t} = \frac{H(t)}{\langle N(t), \hat{z} \rangle} \hat{z} \), which is obviously equivalent to MCF, in the sense of Equation (87). On the other hand, consider a shape parameterized as an "image function," i.e., \( S = (u, v, f(u, v)) \), where the image is given by \( f(u, v) \), and \( u, v \) are a 2D Euclidean domain discretized as the pixel grid. In this case, they show that plugging Equation (86) in to the Beltrami flow is equivalent to the image TV-flow.

Remark: Equivalence by Equation (87) does not account for parameterization, which may induce implicit constraints, as is the case here: During Beltrami flow, the evolving shape's points are constrained to move in the \( \hat{z} \) direction, thus keeping the parameterization \( S = (u, v, f(u, v)) \), contrary to MCF, where no such constraint exists.

This is the reason MCF theoretically converges to a point, while image TV flow converges to a plane \((u, v, \text{const})\).

REFERENCES

- Jonathan Allalo, Ron Kimmel, and Dan Raviv. 2013. Scale invariant geometry for non-rigid shapes. SIAM J. Imag. Sci. 6, 3 (2013), 1579–1597.
- F. Andrea, C. Ballester, V. Caselles, and J. M. Mazón. 2001. Minimizing total variation flow. Different. Integr. Equat. 14, 3 (2001), 321–360.
- Jean-François Aujol, Guy Gilboa, Tony Chan, and Stanley Osher. 2006. Structure-texture image decomposition—modeling, algorithms, and parameter selection. Int. J. Comput. Vis. 67 (2006), 111–136.
- G. Bellettini, V. Caselles, and M. Novaga. 2002. The total variation flow in \( \mathbb{R}^N \). J. Diff. Equat. 184, 2 (2002), 475–525.
- Matania Ben-Artzi and Philippe G. LeFloch. 2007. Well-posedness theory for different. Int. J. Imag. Sci.
- M. Benning, M. Möller, R. Z. Nusek, M. Burger, D. Cremers, G. Gilboa, and C.-B. Schönlieb. 2017. Nonlinear spectral image fusion. In Proceedings of the 6th International (SSVM’17). 41–53.

Shai Biton and Guy Gilboa. 2022. Adaptive anisotropic total variation. Analysis and experimental findings of nonlinear spectral properties. Journal of Mathematical Imaging and Vision 64, 8 (2022), 916–938.

Simon Blatt. 2009. A singular example for the Willmore flow: Analysis 29, 4 (2009), 407–430. DOI: 10.1525/ody.2009.1017. ©Oldenburg, Wissenschaftsverlag, München 2009.

Mario Botsch and Olga Sorkine. 2007. On linear variational surface deformation methods. IEEE Trans. VisuAliz. Comput. Graph. 14, 1 (2007), 213–230.

Amit Bracha, Oshri Hulim, and Ron Kimmel. 2020. Shape correspondence by aligning scale-invariant LBO eigenfunctions. In The Proceedings of the Eurographics Workshop on 3D Object Retrieval (3DOR’20). The Eurographics Association, 1–9.

Alex Bronstein, Yoni Choukroun, Ron Kimmel, and Matan Sela. 2016. Consistent discretization and minimization of the l1 norm on manifolds. In Proceedings of the 4th International Conference on 3D Vision (3DV’16). IEEE, 435–440.

Alexander M. Bronstein, Michael M. Bronstein, and Ron Kimmel. 2008. Numerical Geometry of Non-rigid Shapes. Springer Science & Business Media.

Leon Bungert and Martin Burger. 2020. Asymptotic profiles of nonlinear homogeneous evolution equations of gradient flow type. J. Evolut. Equat. 20, 3 (2020), 1061–1092.

Leon Bungert, Martin Burger, Antonin Chambolle, and Matteo Novaga. 2021. Nonlinear spectral decompositions by gradient flows of one-homogeneous functionals. Analysis & PDE 14, 3 (2019), 823–860.

Martin Burger, Guy Gilboa, Michael Moeller, Lina Eckhardt, and Daniel Cremers. 2016. Spectral decompositions using one-homogeneous functionals. SIAM J. Imag. Sci. 9, 3 (2016), 1374–1408.

M. Burger, G. Gilboa, S. Osher, and J. Xu. 2006. Nonlinear inverse scale space methods. Commun. Math. Sci. 4, 1 (2006), 179–212.

Martin Burger and Stanley Osher. 2013. A guide to the TV zoo. In Level Set and PDE based Reconstruction Methods in Imaging, Springer, 1–70.

Simone Cammarasana and Giuseppe Patané. 2021. Localised and shape-aware functions for spectral geometry processing and shape analysis: A survey & perspectives. Comput. Graph. 97 (2021), 1–18.

Antonin Chambolle. 2004. An algorithm for total variation minimization and applications. J. Math. Imag. Vis. 20, 1 (2004), 89–97.

Antonin Chambolle, Vincent Caselles, Daniel Cremers, Matteo Novaga, and Thomas Pock. 2010. An introduction to total variation for image analysis. Theoret. Found. Numer. Math. Sparse Recon. 9, 263–340 (2010), 227.

Antonin Chambolle and Thomas Pock. 2011. A first-order primal-dual algorithm for convex problems with applications to imaging. J. Math. Imag. Vis. 40, 1 (2011), 120–145.

Paolo Cignoni, Roberto Scopigno, and Marco Tarini. 2005. A simple normal enhancement technique for interactive non-photorealistic renderings. Comput. Graph. 29, 1 (2005), 125–133.

Ido Cohen and Guy Gilboa. 2020. Introducing the p-Laplacian spectrum. Sig. Process. 167 (2020), 107281.

Keenan Crane, Ulrich Pinkall, and Peter Schröder. 2013. Robust fairing via conformal curvature flow. ACM Trans. Graph. 32, 4 (2013), 1–10.

Mathieu Desbrun, Mark Meyer, Peter Schröder, and Alan H. Barr. 1999. Implicit fairing of irregular meshes using diffusion and curvature flow. In Proceedings of the 26th Annual Conference on Computer Graphics and Interactive Techniques. 317–324.

Julie Digne. 2012. Similarity based filtering of point clouds. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops. IEEE, 73–79.

Chinthaka Dinesh, Gene Chen, and Ivan V. Bajic. 2019. 3D point cloud super-resolution via graph total variation on surface normals. In Proceedings of the IEEE International Conference on Image Processing (ICIP’19). IEEE, 4390–4394.

Chinthaka Dinesh, Gene Chen, and Ivan V. Bajic. 2020. Super-resolution of 3D color point clouds via fast graph total variation. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP’20). IEEE, 1983–1987.

Manfred P. de Carro. 2016. Differential Geometry of Curves and Surfaces: Revised and Updated Second Edition. Courier Dover Publications.

Puhsong Duan, Xudong Kang, Shustao Li, and Pedram Chamani. 2019. Noise-robust hyperspectral image classification via multi-scale total variation. IEEE J. Select. Topics Appl. Earth Observ. Rem. Sens. 12, 6 (2019), 1948–1962.

Abhishek Dinoor, Olivier Lezoray, and Sebastien Bougleux. 2008. Nonlocal discrete regularization on weighted graphs: A framework for image and manifold processing. IEEE Trans. Image Process. 17, 7 (2008), 1047–1066.

Marco Fumero, Michael Möller, and Emanuele Rodolà. 2020. Nonlinear spectral geometry processing via the TV transform. ACM Trans. Graph. 39, 6 (2020), 1–16.

Sylvester Gallot, Dominique Hulin, and Jacques Lafontaine. 1990. Riemannian Geometry. Vol. 2. Springer.

Sylvester Gallot, Dominique Hulin, and Jacques Lafontaine. 2004. Differential manifolds. In Riemannian Geometry. Springer, 1–49.

Guy Gilboa. 2013. A spectral approach to total variation. In Proceedings of the International Conference on Scale Space and Variational Methods in Computer Vision, Springer, 36–47.
