Volume 35 (2016), Number 2 pp. 1–14 COMPUTER GRAPHICS *forum*

Spectral Processing of Tangential Vector Fields

Christopher Brandt Leonardo Scandolo Elmar Eisemann Klaus Hildebrandt

Delft University of Technology

Abstract

We propose a framework for the spectral processing of tangential vector fields on surfaces. The basis is a Fourier-type representation of tangential vector fields that associates frequencies with tangential vector fields. To implement the representation for piecewise constant tangential vector fields on triangle meshes, we introduce a discrete Hodge–Laplace operator that fits conceptually to the prominent cotan *discretization of the Laplace–Beltrami operator. Based on the Fourier representation, we introduce schemes for spectral analysis, filtering and compression of tangential vector fields. Moreover, we introduce a spline-type editor for modeling of tangential vector fields with interpolation constraints for the field itself and its divergence and curl. Using the spectral representation, we propose a numerical scheme that allows for real-time modeling of tangential vector fields.*

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—

1. Introduction

Tangential vector fields are a fundamental representation of directional information on a surface. For example, gradients of functions are tangential vector fields and flows, stresses, strains, or curvature directions are tangential to a surface. Therefore, the processing of tangential vector fields is important for algorithms for many applications in graphics.

Spectral methods are well established for the processing of functions and signals over planar domains. Within the last decade, spectral methods for the processing of curved surfaces and functions on them have been developed and the field of spectral mesh processing has been established. Central is the eigendecomposition of the Laplace–Beltrami operator, which generalizes the Fourier basis from planar domains to curved surfaces. Whereas on planar domains, spectral methods can be directly generalized to the processing of vector fields by simply applying the method to each of the component functions of the vector field, this is not possible for tangential vector fields on curved surfaces because there is no rigid Cartesian coordinate system.

In this paper, we introduce a framework for spectral processing of tangential vector fields on curved surfaces. The foundation is a Fourier-type representation that associates frequencies with tangential vector fields. To construct the Fourier-type basis on the space of tangential vector fields, we combine the eigendecomposition of the Hodge–Laplace operator and the Hodge decomposition to obtain basis fields that are either integrable, co-integrable or harmonic.

We formulate the spectral processing framework for piecewise constant tangential vector fields on triangle meshes. These vector fields are widely used in graphics applications and a discrete Hodge decomposition has been established. To define the spectral basis, we introduce a discrete Hodge–Laplace operator for piecewise constant vector fields on surface meshes. The operator shares important properties with its continuous counterpart and fits conceptually to the prominent *cotan* discretization of the Laplace–Beltrami operator. We show that a sparse matrix representation of the operator can be obtained by combining a set of simple matrices. Additionally, we derive a discrete Dirichlet energy that can be used as a regularizer or fairness energy for piecewise constant tangential vector fields on surfaces. For solving the eigenproblem of the discrete Hodge–Laplace operator, we introduce a scheme that boils the computation of the eigenfields down to the computation of the eigenfunctions of two discrete Laplace–Beltrami operators.

To illustrate the potential of our framework for the applications, we introduce tools for spectral filtering and analysis, compression and real-time tangential vector field design. The filtering tools allow users to design filters in the spectral domain. Individual filters can be specified for the integrable and co-integrable parts of a field. The compression scheme allows for lossy compression of tangential vector fields at high compression rates. For vector field design, we follow the approach introduced by Fisher et al. [Fisher2007]. A tangential vector field is constructed by minimizing the Dirichlet energy subject to soft constraints that implement the user input. We demonstrate that by restricting the design space to the space spanned by 1-2k low-frequency eigenfields, the computation is significantly accelerated (up to a factor of 200 in our experiments).

Our second main contribution is a "spline-like"-editor for tan-

gential vector fields. It allows for modeling tangential vector fields using hard constraints on the field and its divergence and curl. Tangential vector field splines (TVFS) can be defined (analogous to cubic splines) as the minimizers of a biharmonic energy under constraints. This idea was already discussed in [Fisher2007] as a possible extension of the design method introduced in the paper. However their approach has two limitations: only soft constraints can be imposed and the resulting scheme is not fast enough to allow for interactive editing. In this work, we overcome both limitations and transfer the approach to piecewise constant vector fields. To be able to formulate TVFS for piecewise constant vector fields, we introduce a biharmonic energy for these fields. Furthermore, we introduce a numerical scheme for computing the TVFS in real-time. The scheme combines the spectral basis and an efficient solver of the quadratic problem. The supplementary video illustrates the benefits of hard constraints for vector field modeling. A live editing session demonstrates the benefit of the TVFS editor for the design of fur on meshes.

We want to remark that after the submission of this paper we found that concurrently to our work the discrete Hodge–Laplace operator for piecewise constant vector fields was also introduced in [dGDT15] and a scheme for computing the eigenfields using eigenfunctions was introduced in [LMH*15], also concurrent to this work.

2. Related Work

Spectral mesh processing Within the last decade a large number of methods for tasks in shape analysis and processing that use the spectrum and eigenfunctions of the Laplace–Beltrami operator have been proposed. In the following, we briefly discuss some of the applications. For an introduction to spectral mesh processing and a broader overview of applications, we refer to the surveys by Lévy and Zhang [LZ09] and Zhang, van Kaick and Dyer [ZvKD10].

The eigendecomposition of the Laplace–Beltrami operator yields a basis of the space of functions on a curved surface analogous to the Fourier basis for signals over a one-dimensional domain. By transforming a function into this basis, it is decomposed into oscillations that are ordered by their frequencies. Even the surface itself (*i.e.*, the embedding of the surface in $\mathbb{R}^3$) can be treated in this way. Vallet and Lévy [VL08] proposed a framework for the design of mesh filters. The framework allows for amplifying and suppressing the different frequencies in the embedding of the mesh. For example, high and low pass filters for sharpening and smoothing the mesh can be constructed. Karni and Gotsman [KG00] proposed a scheme for the compression of triangle meshes based on the eigendecomposition of a combinatorial Laplacian. Recently, a scheme for the compression of dynamic mesh sequences was introduced by Váša et al. [VMHB14]. The regular pattern of minima and maxima of the eigenfunctions are used for generating a quadrangular mesh on a surface. Dong et al. [DBG*06] introduce a technique that uses the Morse–Smale complex of a carefully chosen eigenfunction to generate a coarse quadrangulation. Huang et al. [HZM*08] and Ling et al. [LHJ*14] extended this approach such that it can provide a user with control of the shape, size, orientation, and feature alignment of the faces of the resulting quadrangulation. Spectral methods for shape segmentation have been introduced by Sharma et al. [SHKvL09] and Huang et al. [HWAG09].

The eigenfunctions enjoy many desirable properties. They are intrinsic, which means they do not change if the mesh is isometrically deformed. Since they are derived as the discretization of a continuous concept, they are mesh independent. They are variational, which makes them robust to remeshing. These properties make them well-suited for the design of pose-independent and mesh-independent shape descriptors. Various descriptors have been designed including the *Shape-DNA* introduced by Reuter et al. [RWP05, RWP06], Rustamov's *Global Point Signature (GPS)* [Rus07], the *Heat Kernel Signature (HKS)* proposed by Sun et al. [SOG09], the *Auto Diffusion Function* introduced by Gebal et al. [GBAL09] and the *Wave Kernel Signature* by Aubry et al. [ASC11]. A signature involving not only intrinsic, but also extrinsic information about the surface was introduced in [HSvTP10, HSvTP12]. Based on the GPS and the HKS, Ovsjanikov et al. [OSG08, OMMG10] introduced schemes for the detection of shape symmetries. The pose and mesh independence has also been the basis of schemes for shape correspondences and matching. Dey et al. [DLL*10] propose a robust pose-oblivious shape matching algorithm based on persistent extrema of the HKS. The functional maps framework, introduced by Ovsjanikov at el. [OBCS*12], uses low-frequency eigenfunctions of the Laplace–Beltrami operator on two shapes for constructing a linear map between the function spaces of the shapes. Rustamov et al. [ROA*13] used this functional correspondence to compare shapes.

Discrete Laplace–Beltrami operators are the backbone of spectral mesh processing. Commonly, piecewise linear and continuous functions (linear Lagrange finite elements) over triangle meshes are considered. The discretization of the Laplace–Beltrami operator in this setting leads to the prominent *cotan* weights [PP93, MDSB03]. Discretizations with higher-order elements were consider by Reuter et al. [RWP05]. Properties of the discrete Laplace operators have been studied by Wardetzky et al. [WMKG07, AW11]. Convergence of the discrete operators and their spectrum have been studied by Hildebrandt et al. [HPW06, HP11] and Dey et al. [DRW10].

In this paper, we are proposing a framework that allows for applying spectral methods for the processing of tangential vector fields on surface meshes. The spectral analysis of vector fields on planar domains has been treated by Wagner, Garth and Hagen [WGH12] and a construction of reduced bases for fluid simulation on planar domains using Laplace eigenfunctions was introduced by de Witt, Lessig and Fiume [DWLF12]. These techniques however do not carry over to curved surfaces since they require a fixed Cartesian coordinate system, which is not available for tangential fields on curved surfaces.

Tangential vector field processing Tangential vector fields appear in many applications in graphics. They are used for controlling anisotropic shading of surfaces [MRMH12, RGB*14], non-photorealistic rendering [HZ00, YCLJ12, CYZL14], texture generation [WL01, KCPS15], simulation of fluid and liquids on surfaces [AWO*14, AVW*15], surface segmentation [SBCBG11, ZZCJ14] and surface construction [IBB15, PLS*15]. For a recent surveys on direction field synthesis, design, and processing, we refer to [dGDT15, VCD*16]. Methods that use tangential vector fields

and more general direction fields for surface meshing have received much attention in recent years. Some examples are [RLL*06, KNP07, BZK09, LLZ*11, TPP*11, ECBK14, CK14, LLW15]. For a recent survey on the topic, we refer to [BLP*13].

The Hodge decomposition of vector fields is an important tool for the processing of tangential vector fields. It allows for decomposing the fields into an integrable, a co-integrable and a harmonic part. A discrete Hodge decomposition of the space of piecewise constant vector fields on a surface mesh has been introduced by Polthier and Preuss [PP00, PP03]. Tong et al. [TLHD03] generalized this decomposition to 3-dimensional domains and introduced a multi-resolution representation of vector fields using the potential and co-potential of a vector field. Wardetzky [War06] extended the approach from simply-connected domains to surfaces of arbitrary genus and proved convergence of the decomposition. The spectral decomposition we are proposing is compatible with this discrete Hodge decomposition. For a recent survey on discretizations and applications of the Hodge decomposition, we refer to [BNPB13].

Fairness energies are used for the reconstruction, design and synthesis of tangential vector and direction fields. Fairness energies for different representations of vector and direction fields have been proposed [ZMT06, ZHT07, RVLL08, BZK09, KCPS13, DVPSH14]. Here, we introduce a discretization of the Dirichlet energy for piecewise constant tangential vector fields using a combination of conforming and non-conforming discrete divergence and curl operators.

Alternatively to working with vector fields, one can dualize and consider 1-forms. *Discrete Exterior Calculus* [DHLM05] provides notions of discrete k-forms and discrete operators on them including a discrete Hodge Laplacian for k-forms. A discrete Hodge decomposition for 1-forms was introduced by Fisher et al. [FSDH07] and a spectral decomposition of a discrete Hodge Laplacian on spaces of discrete k-forms has been derived by Arnold et al. [AFW06].

3. Background: Hodge Decomposition of Vector Fields

In this section, we review the Hodge decomposition of vector fields on a smooth surface $\mathcal{M}$ embedded in $\mathbb{R}^3$. We denote the surface normal field by N, and, for any point $p \in \mathcal{M}$ the tangent plane of $\mathcal{M}$ at p by $T_p\mathcal{M}$. Before stating the Hodge decomposition, we introduce basic differential operators on the space of functions and vector fields on surfaces.

Gradient, divergence, curl and Laplace–Beltrami The *gradient* is a linear operator mapping differentiable functions to tangential vector fields. For any point $p \in \mathcal{M}$ the gradient of f at p is defined as the tangential vector $\operatorname{grad} f(p)$ that satisfies

$$\langle \operatorname{grad} f(p), \mathbf{v} \rangle_{\mathbb{R}^3} = \mathrm{d}_{\mathbf{v}} f(p)$$

for all $\mathbf{v} \in T_p\mathcal{M}$. Here, $\mathrm{d}_{\mathbf{v}} f(p)$ is the derivative of f at p in direction $\mathbf{v}$. At any point, $\operatorname{grad} f$ points in direction of the steepest ascent of f. We denote by J the operator that rotates any vector of a vector field in its tangent plane by $\frac{\pi}{2}$ (following the orientation of the surface). Besides the gradient operator, we consider the operator $\mathrm{J}\operatorname{grad}$, which concatenates the gradient and the rotation J. We call this operator the *co-gradient*.

The *divergence* and *curl* are linear operators that map vector fields to functions. The divergence of a vector field $\mathbf{v}$ at a point $p \in \mathcal{M}$ is defined by

$$\operatorname{div} \mathbf{v}(p) = \sum_{i=1}^{2} \langle \nabla_{\mathbf{e}_i} \mathbf{v}(p), \mathbf{e}_i \rangle,$$

where ∇ is the covariant derivative and $\{\mathbf{e}_1, \mathbf{e}_2\}$ forms an orthogonal basis of the tangent plane at p. We define the *curl* as

$$\operatorname{curl} \mathbf{v} = -\operatorname{div} \mathrm{J}\, \mathbf{v}. \tag{1}$$

The divergence and curl are related to the gradient and co-gradient. To describe this relation, we use the L^2-scalar products on the spaces of square-integrable functions and vector fields. These are defined as

$$\langle f, g \rangle_{L^2} = \int_{\mathcal{M}} f(p) g(p) \mathrm{d}A \tag{2}$$

$$\langle \mathbf{v}, \mathbf{w} \rangle_{\boldsymbol{L}^2} = \int_{\mathcal{M}} \langle \mathbf{v}(p), \mathbf{w}(p) \rangle_{\mathbb{R}^3} \mathrm{d}A \tag{3}$$

The divergence and gradient as well as the curl and the co-gradient satisfy

$$\langle \operatorname{div} \mathbf{v}, f \rangle_{L^2} = -\langle \mathbf{v}, \operatorname{grad} f \rangle_{\boldsymbol{L}^2} \tag{4}$$

$$\langle \operatorname{curl} \mathbf{v}, f \rangle_{L^2} = -\langle \mathbf{v}, \mathrm{J} \operatorname{grad} f \rangle_{\boldsymbol{L}^2}, \tag{5}$$

for all pairs of a continuously differentiable function f and tangential vector field $\mathbf{v}$, which follows from an integration by parts. By combining the operators introduced above, we can define the *Laplace–Beltrami* operator

$$\Delta = -\operatorname{div} \operatorname{grad} \tag{6}$$

on the space of twice differentiable functions. Note that the operator could alternatively be defined as the negative of the curl of the co-gradient: $\Delta = -\operatorname{curl} \mathrm{J} \operatorname{grad}$.

Hodge decomposition and harmonic fields The space $\mathcal{X}$ of smooth tangential vector fields on a surface with vanishing boundary can be decomposed into three L^2-orthogonal subspaces

$$\mathcal{X} = \operatorname{Image}(\operatorname{grad}) \oplus \operatorname{Image}(\mathrm{J} \operatorname{grad}) \oplus \mathcal{H},$$

The first subspace is formed by the gradients of smooth functions. Fields in this space are curl-free and represent the integrable part of a vector field. Similarly, the second subspace is the space of co-gradients of smooth functions, which are divergence-free. This space represents the co-integrable part of a vector field. The third space consists of the harmonic tangential vector fields $\mathcal{H}$. These fields are neither gradients nor co-gradients of functions. The space $\mathcal{H}$ can be defined as the intersection of the kernel of the divergence and the curl. In other words, these are exactly the fields that have vanishing divergence and curl. The space contains information about the topology of the surface. $\mathcal{H}$ equals the first singular cohomology of the surface. This is an important relation between vector calculus and algebraic topology. We refer to the literature, *e.g.* [War13], for more about this connection. One consequence is that the dimension of $\mathcal{H}$ on a surface of genus g is $2g$.

4. Discrete Hodge Decomposition

In graphics applications, we are often dealing with piecewise constant vector fields on triangle meshes. A discrete counterpart of the

Hodge decomposition for piecewise constant vector fields has been introduced in [PP00, PP03, TLHD03, War06]. In this section, we review this construction.

Function spaces We denote the space of vector fields that are constant in every triangle of a mesh $\mathcal{M}_h$ by $\mathcal{X}_h$. In addition, we consider two function spaces. Both consist of functions on $\mathcal{M}_h$ that are linear polynomials in every triangle. The two spaces are constructed by imposing continuity constraints on the linear polynomials of neighboring triangles: the space S_h of piecewise linear polynomials that are globally continuous and the space S_h^* of piecewise linear polynomials that are continuous at the midpoints of all interior edges. The combination of S_h and S_h^* is needed for the discrete Hodge decomposition.

Discrete operators The gradients of functions in S_h and S_h^* are defined for all points in the interior of a triangle and they are constant within each triangle. Hence the gradient is a linear map from S_h and S_h^* into $\mathcal{X}_h$. While the gradient can be directly transferred to the discrete setting, we define the discrete divergence and curl operators indirectly using the gradient and equations (4) and (5). As in the continuous case, the discrete divergence and curl map vector fields to functions. Since we are working with two function spaces, we get two discrete divergence and curl operators. The conforming discrete divergence is the linear operator

$$\mathrm{div}_h : \mathcal{X}_h \mapsto S_h$$

that satisfies

$$\langle \mathrm{div}_h \mathbf{v}, f \rangle_{L^2} = -\langle \mathbf{v}, \mathrm{grad}\, f \rangle_{\boldsymbol{L}^2}$$

for all $\mathbf{v} \in \mathcal{X}_h$ and $f \in S_h$ and the nonconforming discrete divergence is the linear operator

$$\mathrm{div}_h^* : \mathcal{X}_h \mapsto S_h^*$$

that satisfies

$$\langle \mathrm{div}_h^* \mathbf{v}, g \rangle_{L^2} = -\langle \mathbf{v}, \mathrm{grad}\, g \rangle_{\boldsymbol{L}^2}$$

for all $\mathbf{v} \in \mathcal{X}_h$ and $g \in S_h^*$. Following the definition (1) of the curl in the continuous case, the conforming and nonconforming discrete curl operators are defined as

$$\mathrm{curl}_h \mathbf{v} = -\mathrm{div}_h \mathrm{J}\,\mathbf{v} \quad \text{and} \quad \mathrm{curl}_h^* \mathbf{v} = -\mathrm{div}_h^* \mathrm{J}\,\mathbf{v}.$$

We discuss explicit matrix representations of the operators in Section 5.

Finally, we want to remark that there is a difference between the discrete operators we define here and the definitions in [PP03, War06]. They define the divergences and curls of vector fields as integrated quantities, while in the presented definitions, the divergences and curls of vector fields are piecewise linear functions (hence pointwise quantities). The notions are related, one can use the mass matrix (see Section 5) to convert one into the other. We refer to [WBH*07] for a discussion of integrated vs. pointwise quantities.

Discrete Hodge decomposition As in the continuous case, the discrete Hodge decomposition divides the space of vector fields into three orthogonal subspaces: the image of the gradient, the image of the co-gradient and a space of harmonic vector fields. To obtain spaces of harmonic fields that are $2g$-dimensional as in the continuous case, we need to combine the spaces S_h and S_h^*. The discrete Hodge decomposition of the space of piecewise constant vector fields is

$$\mathcal{X}_h = \mathrm{Image}(\mathrm{grad}_{|S_h}) \oplus \mathrm{Image}(\mathrm{J}\,\mathrm{grad}_{|S_h^*}) \oplus \mathcal{H}_h. \tag{7}$$

The first component, $\mathrm{Image}(\mathrm{grad}_{|S_h})$, consists of the vector fields that are gradients of functions in S_h. This part describes the integrable part of a vector field, which has vanishing *nonconforming* discrete curl. This means the first component is part of the kernel of curl_h^*. The second component, $\mathrm{Image}(\mathrm{J}\,\mathrm{grad}_{|S_h^*})$, consists of the vector fields that are co-gradients of functions in S_h^*, the co-integrable part. These vector fields have vanishing *conforming* discrete divergence. Hence, the second component is part of the kernel of div_h. For surfaces with genus zero (homeomorphic to a sphere), $\mathrm{Image}(\mathrm{grad}_{|S_h})$ is exactly the kernel of curl_h^* and $\mathrm{Image}(\mathrm{J}\,\mathrm{grad}_{|S_h^*})$ is exactly the kernel of div_h. However, for surfaces with non-vanishing genus g there is a $2g$-dimensional space $\mathcal{H}_h$ of piecewise constant vector fields for which curl_h^* and div_h vanish. These vector fields are neither gradients of functions in S_h nor co-gradients of functions in S_h^*. We call

$$\mathcal{H}_h = \mathrm{Kernel}(\mathrm{div}_h) \cap \mathrm{Kernel}(\mathrm{curl}_h^*)$$

the space of *discrete harmonic* vector fields.

We want to emphasize that this theory requires the interplay of the conforming and nonconforming spaces and operators. For example, a vector field is the gradient of a function in S_h (on a simply-connected domain) if curl_h^* vanishes. Moreover, to get spaces of discrete harmonic vector fields of dimension $2g$, the spaces S_h and S_h^* have to be combined. If only gradients and co-gradients of functions in S_h are used, the dimension of the resulting space of harmonic fields depend on the mesh (and not only on the genus of the surface) and are usually large (*e.g.* the dimension grows under mesh refinement).

As an alternative to (7), we could exchange the roles of S_h and S_h^* and obtain the decomposition $\mathcal{X}_h = \mathrm{Image}(\mathrm{grad}_{|S_h^*}) \oplus \mathrm{Image}(\mathrm{J}\,\mathrm{grad}_{|S_h}) \oplus \mathcal{H}_h^*$. Since in this case the integrable part of a vector field is the gradient of a function in S_h^*, we call this the *nonconforming* discrete Hodge decomposition. The resulting space of *nonconforming* discrete harmonic fields is $\mathcal{H}_h^* = \mathrm{Kernel}(\mathrm{div}_h^*) \cap \mathrm{Kernel}(\mathrm{curl}_h)$. The fact that there are two different discrete Hodge decompositions is specific to the discrete setting and is not present in the continuous case. In [War06], it was shown that both decompositions converge to their smooth counterpart under suitable refinement of the surface meshes. In this sense, the two decompositions are similar.

In the following, we will consider only the decomposition (7). However, for any notion we introduce, there is a corresponding notion where the roles of the spaces S_h and S_h^* are exchanged.

5. Discrete Hodge–Laplace Operator for Vector Fields

In this section, we introduce a discrete Hodge–Laplace operator for piecewise constant vector fields on surface meshes and the corresponding discrete Dirichlet and biharmonic energies. Before we consider the discrete setting, we first review the smooth setting.

The smooth setting By combining the operators discussed in Section 3, one can construct the *Hodge–Laplace* operator

$$\mathbf{\Delta} = -(\operatorname{grad}\operatorname{div} + \mathrm{J}\operatorname{grad}\operatorname{curl}) \tag{8}$$

on the space of smooth tangential vector fields on a surface. We use the minus sign in (8) to get positive eigenvalues for the Hodge–Laplace operator.

Since we could not find a reference where (8) is stated, we want to put this formula in context with the literature (the supplementary material includes a more detailed derivation). The Hodge Laplacian is usually defined as an operator on smooth k-forms on a Riemannian manifold [AF02]. For surfaces, 0- and 2-forms can be identified with functions and the Hodge Laplacian for these forms is the Laplace–Beltrami operator (6) for functions. 1-forms on a surface can be identified with vector fields via $\mathbf{v} \leftrightarrow \langle \mathbf{v}, \cdot \rangle$. Using this isomorphism, we can carry over the Hodge Laplacian from 1-forms to vector fields and express this operator in terms of the operators J, grad, div and curl. Formula (8) is the resulting operator. For vector fields on planar domains, this operator agrees with the vector Laplacian. In this sense, the Hodge Laplacian generalizes the vector Laplacian from vector fields on planar domain to tangential vector fields on curved surfaces.

Discrete Hodge–Laplace operators Using the definition (8) of the smooth Hodge–Laplace operator for tangential vector fields on a surface and the discrete operators introduced in Section 4, we can construct the discrete Hodge–Laplace operator for piecewise constant vector fields on a surface mesh. As for the discrete Hodge decomposition, the discretization mixes the conforming discrete divergence and the non-conforming discrete curl operators

$$\mathbf{\Delta}_h = -(\operatorname{grad}\operatorname{div}_h + \mathrm{J}\operatorname{grad}\operatorname{curl}_h^*).$$

The discrete Hodge–Laplace operator $\mathbf{\Delta}_h$ shares many properties with its continuous counterpart $\mathbf{\Delta}$.

1. **Symmetry** The operator $\mathbf{\Delta}_h$ is self-adjoint with respect to the $\boldsymbol{L}^2$-scalar product, *i.e.*,
$$\langle \mathbf{\Delta}_h \mathbf{v}, \mathbf{w} \rangle_{\boldsymbol{L}^2} = \langle \mathbf{v}, \mathbf{\Delta}_h \mathbf{w} \rangle_{\boldsymbol{L}^2}$$
for any pair $\mathbf{v}, \mathbf{w} \in \mathcal{X}_h$.
2. **Harmonic fields** The discrete harmonic vector fields (which we defined in Section 4) are exactly the piecewise constant fields with vanishing discrete Hodge–Laplace operator, *i.e.*,
$$\mathcal{H}_h = \operatorname{Kernel}(\mathbf{\Delta}_h).$$
3. **Positive semi-definite** The operator $\mathbf{\Delta}_h$ is positive semi-definite. The symmetry guarantees that all eigenvalues are real. The harmonic fields have eigenvalue zero, all other eigenvalues are positive.
4. **Locality** The continuous Hodge–Laplace operator is local, *i.e.*, evaluating the smooth Hodge–Laplace of a vector field at a point p does not depend on the surface or the vector field outside of an arbitrarily small neighborhood of p. We achieve this property by using a diagonal mass matrix for the L^2-scalar product on S_h. Then, for any $\mathbf{v} \in \mathcal{X}_h$ and triangle $t \in \mathcal{M}_h$, the vector that $\mathbf{\Delta}_h(\mathbf{v})$ assumes in t depends only on the vectors of $\mathbf{v}$ in the triangles that share a common vertex with t and the geometry of these triangles.
5. **Intrinsic** The discrete Hodge–Laplace operator is an intrinsic operator, *i.e.*, it can be constructed using only the length of all edges of the mesh. As a consequence it does not change if the surface is isometrically deformed.
6. **Hodge decomposition** The discrete Hodge–Laplace operator respects the corresponding discrete Hodge decomposition. The image of an integrable vector field is an integrable vector field and the image of a co-integrable vector field is a co-integrable vector field.

Discrete energies Based on the discrete Hodge–Laplace operator, we introduce two quadratic functionals (or energies) on the space of piecewise constant tangential vector fields: the Dirichlet energy

$$E_{\mathbf{D}}(\mathbf{v}) = \langle \mathbf{\Delta}_h \mathbf{v}, \mathbf{v} \rangle_{\boldsymbol{L}^2} = \int_{\mathcal{M}_h} \left((\operatorname{div}_h \mathbf{v})^2 + (\operatorname{curl}_h^* \mathbf{v})^2 \right) \mathrm{d}A \tag{9}$$

and the biharmonic energy

$$E_{\mathbf{B}}(\mathbf{v}) = \langle \mathbf{\Delta}_h \mathbf{v}, \mathbf{\Delta}_h \mathbf{v} \rangle_{\boldsymbol{L}^2} = \int_{\mathcal{M}_h} \| \mathbf{\Delta}_h \mathbf{v} \|^2 \, dA. \tag{10}$$

For simply-connected surfaces (topological spheres) the energies are positive definite, and, for surfaces of genus $g > 0$, the energies are semi-positive definite. In the latter case, the harmonic fields are in the kernel of the energies. We will use these energies as regularizers for the construction of smooth vector fields.

Matrix representations Matrix representations of all the discrete operators and the energies can be obtained as products and sums of a set of six simple matrices. This illustrates the structure underlying the operators and simplifies the implementation. To get a matrix representation of an operator, we first have to fix bases in the relevant spaces. We use the nodal bases, *i.e.*, a function in S_h is represented by a vector listing a function value for every vertex and a function in S_h^* is represented by a vector listing a function value for every edge. The linear polynomials over each triangle corresponding to a nodal vector are uniquely determined via interpolation. For S_h^*, the nodes are located at the midpoints of the edges. For a vector field in $\mathcal{X}_h$, we are listing one tangential vector for every triangle. To describe the vectors, we fix a (positively oriented) orthonormal basis of the tangent plane in every triangle.

Once the bases are fixed, we can construct the matrices. The first two matrices are the gradients G and G^* on S_h and S_h^*. Both matrices are sparse (three entries per row). Explicit formulas for the computation of the gradient of a linear polynomial over a triangle can be found in [PP93] and [BKP*10, pp. 40–41]. Furthermore, we need three diagonal mass matrices M, M^*, and $\mathbf{M}$ representing the L^2-scalar products in S_h, S_h^* and $\mathcal{X}_h$. The i^{th} diagonal entry of M is a third of the combined area of the triangles adjacent to the i^{th} vertex and the j^{th} diagonal entry of M^* is a third of the combined area of the two triangles sharing the j^{th} edge. Alternatively, the Voronoi areas of the vertices and edge midpoints can be used. The diagonal entries of the matrix $\mathbf{M}$ are simply the areas of the corresponding triangles. We refer to [WBH*07] for a discussion of mass matrices. The last matrix $\mathbf{J}$ is the matrix representation of the operator J that rotates every vector of a piecewise constant vector field by $\pi/2$ in the tangent plane. The matrix is block diagonal, each block consists of a 2×2 matrix that represents the $\pi/2$-rotation in one triangle with respect to the chosen orthonormal basis. All the 2×2

Figure 1: The first, 21st, 51st and 71st pairs of eigenfields on the centaur model. Green fields are curl-free, red fields are divergence-free.

matrices are the same since for any positively oriented orthogonal basis, a $\pi/2$-rotation maps the first entry of the vector to the second and the second to the negative of the first entry.

Matrix representations of all the discussed discrete operators can be obtained as products and sums of these six matrices. For the discrete divergence and curl operators, we have the following matrix representations

div_h	curl_h	div_h^*	curl_h^*
$-M^{-1}G^T\mathbf{M}$	$M^{-1}G^T\mathbf{J}\mathbf{M}$	$-M^{*-1}G^{*T}\mathbf{M}$	$M^{*-1}G^{*T}\mathbf{J}\mathbf{M}$

We denote the matrices representing the discrete Dirichlet energy, Hodge–Laplace operator and biharmonic energy by $\mathbf{S}, \mathbf{L}$ and $\mathbf{B}$. The matrices satisfy

$$\mathbf{S} = \mathbf{M}(GM^{-1}G^T - \mathbf{J}G^*M^{*-1}G^{*T}\mathbf{J})\mathbf{M}$$
$$\mathbf{L} = \mathbf{M}^{-1}\mathbf{S}$$
$$\mathbf{B} = \mathbf{L}^T\mathbf{M}\mathbf{L} = \mathbf{S}\mathbf{M}^{-1}\mathbf{S}$$

To derive the first row, we combine the matrix representations of the operators div_h and curl_h^* and form the energy (9). The second row follows by construction because the discrete Hodge–Laplace operator is the self-adjoint operator corresponding to the discrete Dirichlet energy. For the first step in the third row, we use (10), and the second step follows from the formula for $\mathbf{L}$.

We want to remark that the discrete Dirichlet energy for piecewise linear functions and the discrete Laplace–Beltrami operator can also be constructed from these matrices. The matrix S representing the Dirichlet energy of functions is given by $S = G^T\mathbf{M}G$. This is exactly the *cotan* matrix [PP93, MDSB03]. The matrix L representing the discrete Laplace–Beltrami operator is given (analogous to the second row above) by $L = M^{-1}S$.

6. Fourier Representation of Tangential Vector Fields

In this section, we describe a Fourier type representation of tangential vector fields that associates frequencies with the fields in the space $\mathcal{X}_h$. The representation is based on the eigenfields of the Hodge–Laplace operator, which have to be computed numerically. We introduce a scheme for computing the eigenfields of the Hodge Laplacian that reduces the problem to the computation of the eigenfunctions of the conforming and the nonconforming discrete Laplace–Beltrami operators.

Eigenfields of the discrete Hodge–Laplace operator The eigenfields of $\mathbf{\Delta}_h$ are the solutions of the equation

$$\mathbf{\Delta}_h\mathbf{\Phi} = \lambda\mathbf{\Phi}.$$

Instead of solving this eigenproblem directly, we construct the eigenfields using the eigenfunctions of the conforming and nonconforming discrete Laplace–Beltrami operators

$$\Delta_h = -\mathrm{div}_h\mathrm{grad} \quad \text{and} \quad \Delta_h^* = -\mathrm{div}_h^*\mathrm{grad}.$$

These operators are the *cotan*-Laplacians on S_h and S_h^*. Eigenfunctions of these operators are solutions of the problems

$$\Delta_h\phi = \lambda\phi \quad \text{and} \quad \Delta_h^*\psi = \mu\psi,$$

where $\phi \in S_h$ and $\psi \in S_h^*$. The numerical treatment of the eigenproblem for the conforming operator is the basis for spectral geometry processing and is treated in detail in [VL08]. The nonconforming case can be treated in the same way, only the matrices M^* and $S^* = G^{*T}\mathbf{M}G^*$ are used instead of M and $S = G^T\mathbf{M}G$.

The following Lemma summarizes the relation of the eigenfunctions of Δ_h and Δ_h^* and the eigenfields of $\mathbf{\Delta}_h$ and is the basis of our scheme for computing the eigenfields.

Lemma 1 The gradient of any eigenfunction of Δ_h is an eigenfield of $\mathbf{\Delta}_h$ and the co-gradient of any eigenfunction of Δ_h^* is an eigenfield of $\mathbf{\Delta}_h$. The eigenvalues of an eigenfunction and the corresponding eigenfield are the same.

Proof Let ϕ be an eigenfunction of Δ_h with eigenvalue λ. Then,

$$\begin{aligned}\mathbf{\Delta}_h \operatorname{grad}\phi &= -\operatorname{grad}\operatorname{div}_h \operatorname{grad}\phi - \mathrm{J}\operatorname{grad}\operatorname{curl}_h^* \operatorname{grad}\phi \\ &= \operatorname{grad}\Delta_h\phi = \lambda \operatorname{grad}\phi.\end{aligned}$$

This proves the lemma for eigenfunctions of Δ_h. The statement about the co-gradients of eigenfunctions of Δ_h^* is proved in a similar manner. □

Fourier representation We denote the number of vertices, edges and the genus of our mesh by n_v, n_e, and g. The following lemma shows that we can use Lemma 1 to construct an orthonormal basis of $\mathcal{X}_h$.

Lemma 2 Let $\{\phi_0, \phi_1, ..., \phi_{n_v-1}\}$ and $\{\psi_0, \psi_1, ..., \psi_{n_e-1}\}$ be eigenbases of Δ_h and Δ_h^* (where ϕ_0 and ψ_0 are the constant functions), and let $\{\mathbf{\Gamma}_1, \mathbf{\Gamma}_2, ..., \mathbf{\Gamma}_{2g}\}$ be an orthonormal basis of the subspace of harmonic fields $\mathcal{H}_h$. Then, the set $B = \{\mathbf{\Phi}_1, \mathbf{\Phi}_2, ..., \mathbf{\Phi}_{n_v-1}, \mathbf{\Psi}_1, \mathbf{\Psi}_2, ..., \mathbf{\Psi}_{n_e-1}, \mathbf{\Gamma}_1, \mathbf{\Gamma}_2, ..., \mathbf{\Gamma}_{2g}\}$, where $\mathbf{\Phi}_i = \frac{1}{\|\operatorname{grad}\phi_i\|_{\mathbf{L}^2}} \operatorname{grad}\phi_i$ and $\mathbf{\Psi}_i = \frac{1}{\|\operatorname{grad}\psi_i\|_{L^2}} \mathrm{J}\operatorname{grad}\psi_i$, is an orthonormal basis of the space $\mathcal{X}_h$ of piecewise constant tangential vector fields.

Proof We first show that any pair of vector fields from B is orthogonal. For $i \neq j$, we have

$$\left\langle \operatorname{grad}\phi_i, \operatorname{grad}\phi_j \right\rangle_{\mathbf{L}^2} = \left\langle \Delta_h\phi_i, \phi_j \right\rangle_{L^2} = \lambda_i \left\langle \phi_i, \phi_j \right\rangle_{L^2} = 0,$$

which implies

$$\left\langle \mathbf{\Phi}_i, \mathbf{\Phi}_j \right\rangle_{\mathbf{L}^2} = 0.$$

In a similar manner, we can show that any pair $\mathbf{\Psi}_i, \mathbf{\Psi}_j$ is orthonormal. The discrete Hodge decomposition (7) guarantees that any pair of vector fields with different letters is orthogonal, because such fields are in different components of the Hodge decomposition. It remains to show that the number of vector fields in the set equals the dimension of the space $\mathcal{H}_h$. The set B consistes of $|B| = n_v - 1 + n_e - 1 + g$ vector fields. Using the Euler formula $n_v - n_e + n_f = 2 - 2g$, we get $|B| = 2n_e - n_f$. Since our mesh is a closed manifold, every edge is in two triangles, which means $3n_f = 2n_e$. Using this equation, we get $|B| = 2n_f$, which is exacty the dimesion of $\mathcal{X}_h$. We showed that B is an orthonormal set in $\mathcal{X}_h$ with $2n_f$ elements, which proves that B is an orthonormal basis of $\mathcal{X}_h$. □

As consequence of the lemma, we can represent any field $\mathbf{v} \in \mathcal{X}_h$ in the basis B

$$\mathbf{v} = \sum_{i=1}^{n_v-1} \alpha_i \mathbf{\Phi}_i + \sum_{i=1}^{n_e-1} \beta_i \mathbf{\Psi}_i + \sum_{i=1}^{2g} \gamma_i \mathbf{\Gamma}_i, \tag{11}$$

where $\alpha_i = \langle \mathbf{v}, \mathbf{\Phi}_i \rangle_{\boldsymbol{L}^2}$, $\beta_i = \langle \mathbf{v}, \mathbf{\Psi}_i \rangle_{\boldsymbol{L}^2}$ and $\gamma_i = \langle \mathbf{v}, \mathbf{\Gamma}_i \rangle_{\boldsymbol{L}^2}$. Since any basis field is an eigenfield of the discrete Hodge–Laplace operator, we can associate a frequency, the square root of the eigenvalue, to every basis field. Hence, this representation associates frequencies with tangential vector fields. In this sense, (11) is a Fourier representation. In the following sections, we will show benefits of this representation for the applications.

In the continuous case, the eigenfunctions come in pairs. For every eigenfield $\mathbf{\Phi}$, the rotated field, J $\mathbf{\Phi}$, is also an eigenfield with the same eigenvalue. Since we construct the basis as gradients and co-gradients of eigenfunction of the Laplace–Beltrami operator, we choose in every eigenspace the basis such that every basis vector is in one component of the Hodge decomposition. As a result, the Fourier representation refines the Hodge decomposition. Since in the discrete case two different function space, S_h and S_h^*, have to be combined, the symmetry is broken and "pairs" of eigenfields have only approximately the same eigenvalues. Figure 1 shows "pairs" of eigenfields of the discrete operator.

Computation of the eigenfields Our scheme for computing the low-frequency eigenfields proceeds in three steps. The input is a maximum eigenvalue $\lambda_{\max}$ (or alternatively the number of eigenfields to be computed). The first step is to compute a basis of the space of discrete harmonic fields (which along the way provides us with a basis of the cohomology of the surface). The space is $2g$-dimensional, where g is the genus of the surface. The idea is to project $2g$ random vector fields to the space of harmonic fields. These will span the space of harmonic fields (the probability that the random vectors or their projections are linearly dependent vanishes). Finally, we orthonormalize these vectors. To project a vector field $\mathbf{v}$ into the space of harmonic fields, we remove the integrable and the co-integrable parts. The potentials of the integrable and the co-integrable part can be determined by solving the least squares problem

$$\operatorname*{argmin}_{f \in S_h, g \in S_h^*} \|\mathbf{v} - \operatorname{grad} f - \mathrm{J}\operatorname{grad} g\|^2.$$

Since the integrable and co-integrable parts are orthogonal, solving this problem can be carried out in two steps: first compute the integrable part, then the co-integrable part. Both steps require solving a linear system where the matrices are the cotan matrices $S = G^T\mathbf{M}G$ and $S^* = G^{*T}\mathbf{M}G^*$. The second step is to compute linearly independent eigenfunctions ϕ of Δ_h with eigenvalue smaller than $\lambda_{\max}$. Since all eigenvalues are positive, we compute bands of eigenfunctions with increasing eigenvalue starting with zero until we reach $\lambda_{\max}$. Then we compute the corresponding eigenfields $\mathbf{\Phi} = \operatorname{grad}\phi$ and orthonormalize them. The third step is to compute the eigenfunctions ψ of Δ_h^* with eigenvalue smaller than $\lambda_{\max}$. As before, we compute the corresponding eigenfields $\mathbf{\Psi} = \mathrm{J}\operatorname{grad}\psi$ and orthonormalize them. For the computation of the eigenfunctions of Δ_h and Δ_h^*, we follow [VL08].

The resulting eigenbasis refines the discrete Hodge decomposition. By construction each type of eigenfield belongs to one sub-

space of $\mathcal{X}_h$. The fields $\mathbf{\Phi}$ are in the integrable component, the fields $\mathbf{\Psi}$ in the co-integrable component and the rest forms a basis of the space of harmonic fields.

7. Tangential Vector Field Splines

Many classical splines can be characterized by variational principles. Splines in tension are minimizers of a weighted sum of the biharmonic and the Dirichlet energy subject to constraints (cubic splines are the special case when the weight of the Dirichlet energy vanishes). Following the classical example, we define the tangential vector field splines (TVFS) as minimizers of the weighted sum of the biharmonic and the Dirichlet energy

$$E_{\mathbf{B}}(\mathbf{v}) + \omega E_{\mathbf{D}}(\mathbf{v}) \tag{12}$$

subject to linear equality constraints on the vectors of $\mathbf{v}$ and its divergence and curl. Vortices can be constructed by prescribing non-zero curl, sinks and sources are created via positive or negative divergence respectively. Alternatively, singularities can be constructed by specifying a few vector constraints around their desired locations. We refer to the several images and the supplementary video to see several examples of this type of topology control in action. The biharmonic energy is needed to obtain smooth enough vector fields that satisfy the hard constraints. The effect is shown in Figure 2.

We want to remark that the idea of defining tangential vector field splines as minimizers of (12) was introduced in [FSDH07] as an extension of their vector field design approach. However, their approach has two limitations: only soft constraints can be imposed and the resulting scheme is not fast enough to allow for interactive editing.

Real-time computation The computation of a TVFS amounts to solving a sparse quadratic problem with linear equality constraints. The challenge is to solve the problems at real-time rates to enable an interactive TVFS editor.

Directly solving the resulting linear systems to compute a TVFS is not an option since this can take several minutes. Directly reusing a sparse factorization is not possible, because the size of the matrix changes, whenever constraints are added or removed. A fast approximation algorithm can be established using the basis of eigenfields. We are using the property of the eigenbasis that in this basis the Dirichlet and the biharmonic energy are represented by diagonal matrices.

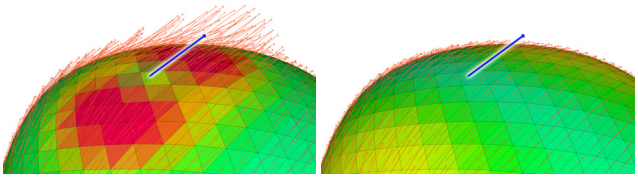

Figure 2: The per-face Dirichlet energy of a tangential vector field spline from a single constraint on an irregular sphere is shown (red: high Dirichlet energy, green: low Dirichlet energy). On the left, we show the minimizer of the Dirichlet energy only, without higher order regularizer, on the right, we use the biharmonic energy (12) with low ω.

Figure 3: Real-time tangential vector field spline editing on the armadillo model (left, 331904 faces) and the resulting fur (right).

In a preprocess, we compute the d eigenfields with the smallest eigenvalue as described in Section 6. We assemble the vectors to form the columns of a matrix $\mathbf{U} \in \mathbb{R}^{2n_f \times d}$. A tangent field in the d-dimensional subspace can be described by reduced coordinates, *i.e.*, a vector $v \in \mathbb{R}^d$. The matrix $\mathbf{U}$ transforms from reduced coordinates v to the full coordinates $x = \mathbf{U}v \in \mathbb{R}^{2n_f}$. Since we are using a basis of eigenfields, in the reduced coordinates the energies are represented by diagonal matrices. The matrix $\mathbf{\Lambda}$ representing the Dirichlet energy has the eigenvalues on the diagonal. Then, $\mathbf{\Lambda}^2$ represents the biharmonic energy and the resulting energy matrix $\mathbf{D}$ is

$$D = \mathbf{\Lambda}^2 + \omega\mathbf{\Lambda}.$$

We consider n_c linear constraints, which in the unreduced coordinates have the form $\tilde{\mathbf{C}}x = c$, where $\tilde{\mathbf{C}} \in \mathbb{R}^{n_c \times 2n_f}$, $c \in \mathbb{R}^{n_c}$ and $x \in \mathbb{R}^{2n_f}$. To constrain the divergence or curl of the field at a vertex or an edge, we copy the corresponding row from the divergence and curl matrices (see Section 5) into $\tilde{\mathbf{C}}$. The entry in the vector c specifies the value the divergence or curl assumes. Values at arbitrary locations in a triangle can be specified using barycentric coordinates. In a similar manner, vectors in triangles can be prescribed. Once the matrix $\tilde{\mathbf{C}}$ is constructed, we can obtain its reduced counterpart by matrix multiplication: $\mathbf{C} = \tilde{\mathbf{C}}\mathbf{U}$. The matrix $\mathbf{C}$ ensures that the reduced solution exactly satisfies the constraints. The matrix is of small size, $\mathbf{C} \in \mathbb{R}^{n_c \times d}$.

Now we describe how to efficiently solve the constrained linear system. We first consider the case of a surface of genus 0. In this case, the matrix $\mathbf{D}$ has full rank and since it is diagonal, it can be easily inverted. Using Lagrange multipliers, represented by a vector $\mu \in \mathbb{R}^{n_c}$, the solution v of the constraint optimization problem is computed by solving the linear system

$$\begin{aligned} \mathbf{D}v - \mathbf{C}^T\mu &= 0 \\ \mathbf{C}v &= c \end{aligned}$$

Instead of solving this system directly, we first transform it. To eliminate v from the second equation, we multiply the first equa-

tion by $\mathbf{C}\mathbf{D}^{-1}$ and subtract it from the second equation

$$\mathbf{D}v - \mathbf{C}^T\mu = 0 \tag{13}$$

$$\mathbf{C}\mathbf{D}^{-1}\mathbf{C}^T\mu = c \tag{14}$$

To compute the solution v, we first solve (14) for μ and then (13) for v. To compute μ, we factor the matrix $\mathbf{C}\mathbf{D}^{-1}\mathbf{C}^T$, which is a very small matrix of size $n_c \times n_c$. A new factorization is computed whenever the set of constraints changes, changing the value of the constraints affects only the right-hand side of the equation. Solving for v is very fast since $\mathbf{D}$ is diagonal.

In the case, of a surface of genus $g > 0$, the matrix $\mathbf{D}$ has the harmonic fields in its kernel. The resulting system can be solved by treating the harmonic part separately. The system can be rearranged such that first the harmonic part and the Lagrange multipliers are determined and then v is computed. An alternative is to slightly modify the system by setting the eigenvalues of the harmonic part to a small positive constant (*e.g.*, a tenth of the lowest non-zero eigenvalue).

Since we are restricting the computation to a subspace spanned by low-frequency fields, the algorithm computes a low-pass filtered TVFS. From a signal theoretic point of view, the TVFS are low-frequency fields by construction. Our computational scheme cuts-off the remaining high-frequencies of the field. In this sense, the reduced solution could even be the preferred solution. Using bases of 1-2k eigenfields, the reduced results are typically very close to the TVFS.

8. Applications and Experiments

8.1. Computation of the eigenfields

Table 1 lists timings for the computation of the eigenfields of the Hodge Laplacian. The column *Bases setup* contains the total time to setup the conforming and nonconforming Laplace–Beltrami operator, computing their eigenfunctions and then computing their gradients and co-gradients. To compute the low-frequency eigenfunctions, we use the shift-and-invert Lanczos method. The implementation was done in Java with native calls to the MUMPS library [ADKL01] for solving the sparse linear systems. The computation was performed on a Dell Precision M3800.

Figure 1 shows examples of low-frequency eigenfields. For each integrable field (green), the corresponding co-integrable field (red) is shown.

8.2. Spline editor for tangential fields

In Section 7, we described a system for modeling tangential vector field splines in real-time, making use of our reduced basis. We implemented this system, using a dense Cholesky factorization and utilizing the GPU to quickly map from the reduced space to a full field representation. In total, we get real-time responses in an interactive editing environment, where vector constraints can be specified via click-and-drag and a globally optimal tangential field is instantly updated. The method scales well with the sizes of the meshes and allows for real-time tangential vector field modeling on larger meshes: aside from mapping the reduced coordinates to the full representation (which can be done efficiently on the GPU), the size of the system to be solved is independent of the resolution of the mesh.

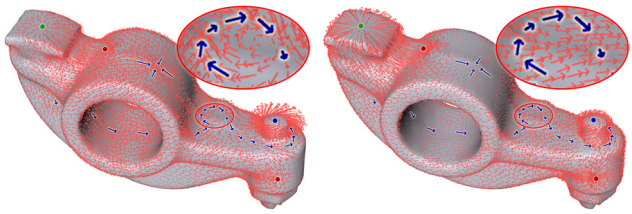

Figure 4: Comparison between editing using our tangential vector field spline editor as described in Section 7 (left) and using soft constrained vector field design (cf. [FSDH07]) (right). As can be seen, not all constraints on the right are satisfactorily obeyed.

In Table 1, 4th column, we list the resulting total time for generating tangential vector field splines on various meshes from 30 user defined constraints using a basis of 1000 eigenfields. We separately list the timings for setting-up and factorizing the system and the timing for a solve (including the time for sending the reduced coordinates to the GPU and there mapping them to the full representation). When a new constraint is added, both those steps have to be executed, but if a constraint is just modified (*e.g.*, changing the direction of a vector constraint), only the solve time has to be taken into account, which is below 5ms even for our largest test mesh (the Armadillo mesh).

In Figure 4, top, we show a result of designing a vector field using our spline editor. As can be seen, all constraints are exactly obeyed but the overall field is still very smooth. In the supplementary video, a real-time editing session can be seen. Another result of TVFS editing on a high-resolution meshes can be seen in Figures 3 and 5, where we effortlessly designed tangent vector fields on meshes with 331k and 70k faces respectively.

8.3. Fur design

As an application of tangential vector field spline editing, we introduce a tool for fur design on surface meshes. A demonstration is shown in the supplementary video and results are shown in Figures 5 and 3. The efficient and intuitive way to design smooth tangent

Model Name	#faces	Bases setup	Spline Editing Setup \Solve	Reduced Soft Design Factor \Solve	Full Soft Design Factor \Solve
Hand	12184	23s	12ms \ 1ms	148ms \ 2.5ms	1.4s \ 16ms
Rocker Arm	20088	43s	15ms \ 1ms	148ms \ 2.6ms	3.3s \ 28ms
Bunny	69666	184s	18ms \ 2ms	155ms \ 2.4ms	22.7s \ 141ms
Bumpy Torus	140240	395s	17ms \ 2ms	169ms \ 3.1ms	86.8s \ 379ms
Armadillo	331904	1246s	49ms \ 4ms	150ms \ 3.5ms	187.7s\ 818ms

Table 1: Timings for tangential vector field bases computation, solving the reduced tangential vector field spline system and solving the reduced and full design system. In all three systems we used a second order energy term as an additional regularizer. In all examples a basis size of 1000 was used (500 divergence free and 500 curl free eigenfields of the Hodge–Laplace operator) and 30 constraints were specified.

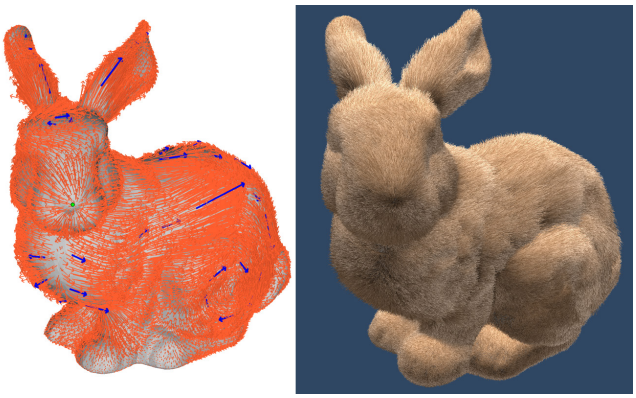

Figure 5: Fur design on the bunny mesh. Left: constraints and resulting tangential vector field spline, right: output field visualized as fur on the bunny.

fields via few hard constraints allows a designer to edit fur on surface meshes in real-time by specifying the length and direction of the hair at certain spots, while aiming for an overall smoothly varying hair direction.

For this type of design task real-time visual responses are crucial, which is made possible with our reduction via the spectral basis. More generally does the reduced basis allow for efficient updates to the GPU when the tangential field changes, which can speed-up the visualization of CPU-run simulations of tangential vector fields.

8.4. Speeding-up soft constrained design

In Fisher et al. [FSDH07], a method for designing vector fields using weak constraints has been proposed. Sinks, sources and vortices can be constructed via prescribing non-zero curl and divergence at specific vertices/edges, but vanishing curl and divergence on the rest of the mesh. Additionally, interpolation constraints for the vector field can be imposed. Since this, in general, over-constrains the system, they are treated as weak constraints and a least-squares problem is solved. As a regularizer, either the Dirichlet energy or a weighted sum of the biharmonic and the Dirichlet energy are used. Denoting the linear constraints by $\tilde{C}x = c$, as in Section 7, and the weight of the biharmonic energy by ω, the linear system

$$(S + \omega B + \tilde{C}^T \tilde{C})x = \tilde{C}^T c$$

has to be solved to construct a field. To solve the system, a sparse Cholesky factorization is computed once and used to solve the systems. When the set of constraints changes, a sparse Cholesky update is computed.

By restricting the construction to a subspace of d =1-2k low-frequency modes, we get a fast approximation algorithm for this system. In particular, the computational cost for the reduced computation depends on the dimension of the subspace and is independent of the resolution of the mesh. The only operations that depend on the mesh size are the mapping of the reduced coordinates to the unreduced coordinates and the construction of the subspace, which is done in a preprocess. Using the notation of Section 7, the reduced system is

$$(\Lambda + \omega\Lambda^2 + C^T C)v = C^T c,$$

which is a d-dimensional system. To solve the system, we compute a dense Cholesky decomposition and use dense Cholesky updates, when the set of constraints changes.

After employing the spectral basis to the resulting system, we are able to speed-up the computation times of the designed tangential fields by a factor of up to 200 (for higher resolution meshes this factor will become larger). Figure 6 shows an example of a reduced and an unreduced solution to the system. One can see that the reduced solution is a smoother field (since additional high frequencies are cut-off). Note that Fisher et al. propose their method in the setting of discrete 1-forms. However, to make a better comparison, we re-implemented their system to work with piecewise constant tangential fields. We solve the resulting sparse linear systems using the MUMPS library, which provides sparse Cholesky factorizations. The speed-up can be observed from the timings listed in Table 1, where one can see, that already for small meshes we gain significantly shorter computation times, and that for large meshes, interactive design is made possible at all only by using our reduced basis. We list timings for factorizations and solving the systems separately. We create tangential fields from 30 user defined constraints and, in the reduced case, we use a basis of 1000 eigenfields. The biharmonic energy was added as a regularizer. Note that adding constraints does not require a re-factorization in either the full or the reduced case, since the factorization can be updated using sparse or dense Cholesky updates, which take less than a millisecond.

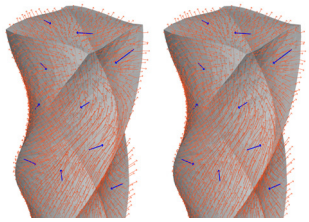

Figure 6: Comparison of unreduced (left) and reduced (right) vector field design.

To highlight the difference of our tangential vector field spline editing and the vector field design system discussed in this section, we point to Figure 4. Here one can see how not all constraints in the least squares system are satisfactorily obeyed. This is also highlighted in a section of our supplementary video. The reason for this is that the user needs to specify the weights for the prescribed vectors and the magnitudes of the prescribed curl and divergence, which results in a trade-off between satisfying the locations of prescribed sinks, sources and vortices versus obeying the prescribed vectors. In the shown example, no weights were found where both types of constraints were obeyed satisfactorily at the same time.

8.5. Spectral analysis and filtering

The Fourier representation (11) discussed in Section 6 allows for spectral analysis and filtering of tangential vector fields. In practice, we only compute the harmonic and first k integrable and co-integrable eigenfields, where k is between 500 and 5000. Then any

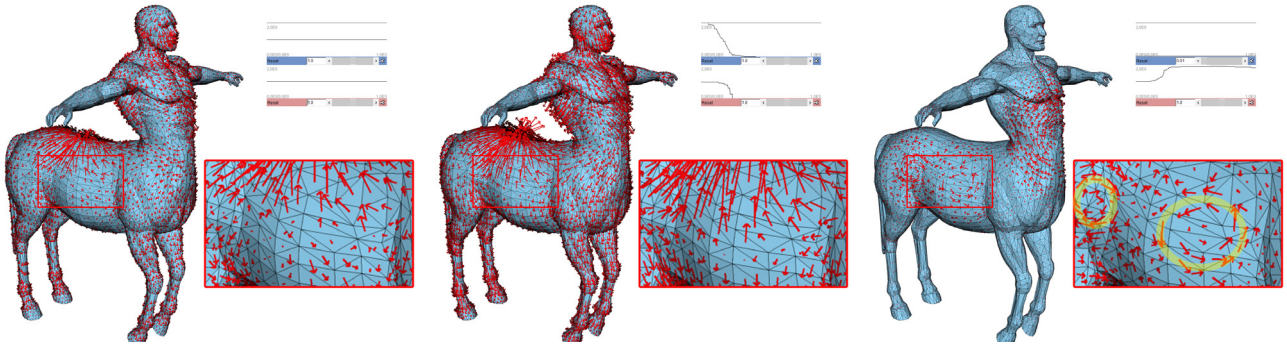

Figure 7: Applying spectral filters to a custom vector field (left). The integrable and co-integrable parts of the field are filtered individually. Two results (middle and right) are shown.

piecewise constant vector field $\mathbf{v}$ can be written as

$$\mathbf{v} = \sum_{i=1}^{k} (\alpha_i \mathbf{\Phi}_i + \beta_i \mathbf{\Psi}_i) + \sum_{i=1}^{2g} \gamma_i \mathbf{\Gamma}_i + \mathbf{v}^{\mathrm{r}} \tag{15}$$

where $\mathbf{v}^r$ is the "rest-field" of $\mathbf{v}$, *i.e.* $\mathbf{v}^{\mathrm{r}} = \sum_{i=k+1}^{n} \alpha_i \mathbf{\Phi}_i + \sum_{i=k+1}^{m} \beta_i \mathbf{\Psi}_i^{\mathrm{c}}$. The coefficients α_i, β_i describe the contribution of $\mathbf{v}$ to the corresponding eigenfields. They are ordered by ascending frequency and can be analyzed and manipulated for spectral processing of the field. Note that the harmonic eigenfields are in the kernel of $\mathbf{\Delta}_h$ and thus contribute to the lowest frequency part of the field.

To enable the spectral processing of tangential vector fields via the creation of spectral filters, we use the established method (cf. [VL08]) of enabling the user to "draw" functions $F_\alpha, F_\beta : \mathbb{R}^+ \to [0, \tau]$, such that a new vector field $\mathbf{v}^*$ is acquired by replacing the spectral coefficients of $\mathbf{v}$ by new coefficients $\alpha_i^* = F_\alpha(\sqrt{\lambda_i}) \cdot \alpha_i$ and $\beta_i^* = F_\beta(\sqrt{\lambda_i}) \cdot \beta_i$. Here, λ_i is the i^{th} distinct eigenvalue of the Hodge Laplacian and τ signifies the largest possible magnification of a coefficient. The reason for scaling the coefficients by $F_{\alpha/\beta}(\sqrt{\lambda_i})$ is that the frequency of the i^{th} eigenfunction is related to the square root of λ_i.
In addition, constants γ_{h} and γ_{r} can be defined, which specify the scaling of the harmonic part of the field and the rest field $\mathbf{v}^{\mathrm{r}}$. A typical "low-pass" filter is $F_{\alpha/\beta}(\lambda) = e^{-\lambda}$ which keeps low frequencies intact while exponentially suppressing high frequencies, which can be used to simplify fields or remove unwanted noise. In the same vein, a "high-pass" filter is $F_{\alpha/\beta}(\lambda) = 2 - e^{-\lambda}$, which can be used to "sharpen" the field. The integrable and co-integrable parts of the field can be filtered separately, which allows for a spectral analysis of the divergence-free and curl-free part of the vector field.

Spectral filtering can be seen in action in our supplementary video as well as in Figure 7, where we show how it can be used to interactively edit and analyze tangential fields. As a first step, we perform a spectral analysis. The plot of the coefficients of the field in the spectral domain for both the integrable and co-integrable eigenfields is shown in Figure 8. The unaltered field is shown on the left of Figure 7. In the magnified area the vectors seem to exhibit noise, and indeed, when applying a low-pass filter on both the field and additionally enhancing the integrable low-frequency part (middle), we get a smoothed and more structured field. Since we magnified only the integrable part, the vortex on the stomach of the centaur disappears. To analyze the noisy features, we enhance the high-frequency part of the field and remove the low frequency integrable part (right). We can see that the high-frequency part mainly consists of two vortices on the side of the centaur (highlighted).

8.6. Compression

The spectral decomposition and the transformation into the frequency domain described above can immediately be used to reduce data rates required to represent tangent vector fields on meshes: instead of storing a vector field on a triangular mesh by specifying two values per triangle (coordinates with respect to an edge or similar common representations), a field can be represented by the $n_v + n_e + 2g$ coefficients $\alpha_i, \beta_i, \gamma_i$. Compression can be achieved by cutting off the higher coefficients, since the high frequency part of the fields is typically small. This approach extends existing approaches from mesh compression [KG00] or dynamic mesh compression [VMHB14] to the compression of tangential vector fields on meshes. Rate-distortion curves can be found in Figure 9.

In Figure 10, we compressed a field exhibiting over 100 small curls on a bunny mesh using the lowest 1k integrable and co-integrable eigenfields. All features are kept faithfully intact.

In Figure 9, left, we plot the relative L^2-error when using a varying number of eigenfields to reproduce the fields on the hand model and on the rocker arm, both shown in Figure 9, right, as well

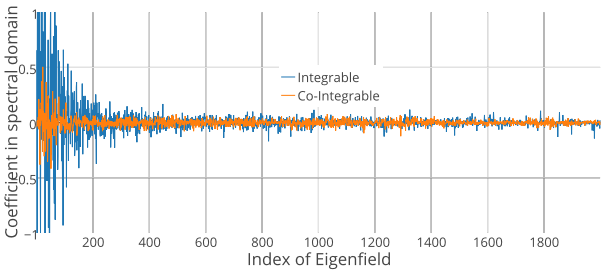

Figure 8: Spectral analysis of the field shown in Figure 7.

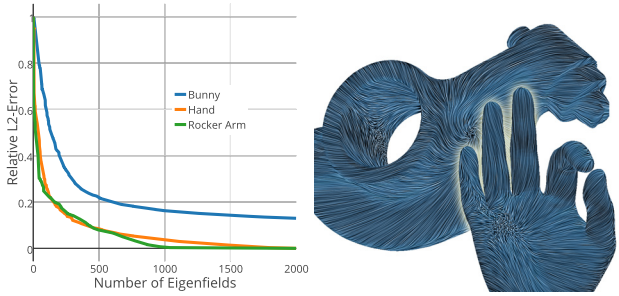

Figure 9: Rate distortion curves when compressing various custom tangential vector fields on three different meshes.

as the field on the bunny shown in Figure 10. The plot is essentially a rate-distortion curve, as the number k of pairs of eigenfields used to compress the field directly relates to the data rate, namely the number of bits per vertex is $2 \cdot 64 \cdot k/n$ (where n is the number of vertices) when using double precision for the coefficients. In case of the tangential field on the bunny mesh, the error only slowly converges to 0, since the field contains a lot of high frequency elements (of course, the error still reaches 0 as the number of eigenfields reaches the number of vertices plus the number of edges). This is visualized in Figure 10, right, where the part of the field is shown which *cannot* be reproduced by the 1k lowest pairs of eigenfields (note that the LIC visualization shows the structure of the field, but does not reveal its magnitude). When creating the same plot for a more regular field, namely the ones shown in Figure 9, we can see that we are able to reach an almost lossless compression by using the 2k lowest pairs of eigenfields. It is worth noting that quite a large number of eigenfields is required to get a low L^2-error, even for this fairly simple field, while a very small number of eigenfields is required to get visually indistinguishable results that preserve all the large features. The compression of the field on the rocker arm yields the best results, where we get an essentially lossless compression for $k = 1100$, which corresponds to a compression rate of 36.52 when comparing to the usual representation of two doubles per triangle.

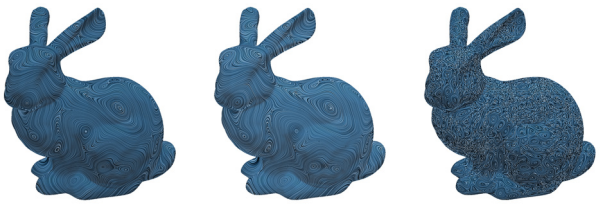

Figure 10: A field exposing a lot of small features (left) and the compressed version made from the 5from the 1000 lowest pairs (middle). In the last picture (right) we show the part of the original field that cannot be constructed from the 1000 lowest pairs of eigenfields.

In Figure 11 we show a snapshot of a time-dependent tangent field on the bumpy torus (140240 faces), consisting of 500 fields, which amounts to a file size of 560,96 Megabyte when representing the vectors in each face by two single precision floating point numbers (4 bytes). We compress the sequence using a basis of 500 eigenfields which amounts to a file-size of exactly 1 Megabyte. On average we get a relative L^2-error of 10 percent between compressed and uncompressed frames, however, as can be seen in the supplementary video, where both time-series are shown next to each other, the two time series are visually indistinguishable.

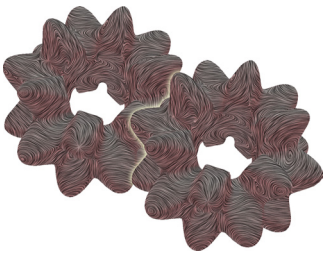

Figure 11: Snapshot from the uncompressed (left) and compressed (right) time-dependent tangential vector field on the bumpy torus.

The timings for the compression are very low, once the eigenfields have been computed, which only needs to be done once per mesh (for timings see Table 1). Whenever a new field on the mesh is to be compressed, all that needs to be done is to compute the $\boldsymbol{L}^2$-scalar products of the field with the $2 \cdot k$ eigenfields (0.67 seconds for the hand-mesh and $k = 1000$). Recovering the usual representation of the vector simply requires a dense matrix vector product of the matrix containing the eigenfields as columns with the coefficient vector (0.06 seconds for the hand-mesh and $k = 1000$, below 1ms when done using the GPU).

9. Conclusion

We introduce a framework for spectral processing of tangential vector fields using a Fourier-type representation of tangential vector fields that associates frequencies with tangential vector fields. To formulate the framework for piecewise constant vector fields on surface meshes, we introduce a discretization of the Hodge–Laplace operator. We demonstrate how techniques from spectral mesh processing can be transferred to tangential vector field processing using this framework. We show results for spectral filtering, analysis and compression. Moreover, we introduce a spline-like editor for modeling tangential vector fields using interpolation constraints. Based on the spectral representation, we propose a computational scheme that enables modeling of tangential vector field splines in real-time.

Future work One direction of future work is to find more applications of the Fourier representation of tangential vector fields, *e.g.*, by transferring techniques from spectral mesh processing to tangential vector field processing.

For applications, *e.g.*, surface meshing, more general types of fields (direction fields, RoSy fields,...) are considered. We are working on establishing a Fourier-type representation for direction fields. This involves the construction of analog differential operators and decompositions for these fields. In this paper, we are for-

mulating the Hodge Laplacian and Dirichlet energy in terms of the classical operators grad, div and curl. This could be helpful since recent work is already addressing the construction of a curl operator [DVPSH15] for direction fields to ensure integrability.

Acknowledgements

The idea of using a random projection algorithm for computing bases of the spaces harmonic vector fields originates from discussions with Max Wardetzky. We would like to thank Mirela Ben-Chen for inspiring discussions and the anonymous reviewers for helpful comments and suggestions. This work was partially supported by the EU project Harvest4D (FP7-323567) and the Intel Visual Computing Institute at Saarland University.

References

[ADKL01] AMESTOY P. R., DUFF I. S., KOSTER J., L'EXCELLENT J.-Y.: A fully asynchronous multifrontal solver using distributed dynamic scheduling. *SIAM Journal on Matrix Analysis and Applications 23*, 1 (2001), 15–41. 9

[AF02] AGRICOLA I., FRIEDRICH T.: *Global Analysis: Differential Forms in Analysis, Geometry, and Physics*. AMS, 2002. 5

[AFW06] ARNOLD D. N., FALK R. S., WINTHER R.: Finite element exterior calculus, homological techniques, and applications. *Acta numerica 15*, 1 (2006). 3

[ASC11] AUBRY M., SCHLICKEWEI U., CREMERS D.: The wave kernel signature: A quantum mechanical approach to shape analysis. In *ICCV* (2011), pp. 1626–1633. 2

[AVW*15] AZENCOT O., VANTZOS O., WARDETZKY M., RUMPF M., BEN-CHEN M.: Functional thin films on surfaces. In *Symposium on Computer Animation* (2015), pp. 137–146. 2

[AW11] ALEXA M., WARDETZKY M.: Discrete Laplacians on general polygonal meshes. *ACM Trans. Graph. 30*, 4 (2011), 102:1–102:10. 2

[AWO*14] AZENCOT O., WEISSMANN S., OVSJANIKOV M., WARDETZKY M., BEN-CHEN M.: Functional fluids on surfaces. In *Computer Graphics Forum* (2014), vol. 33. 2

[BKP*10] BOTSCH M., KOBBELT L., PAULY M., ALLIEZ P., LEVY B.: *Polygon Mesh Processing*. AK Peters, 2010. 5

[BLP*13] BOMMES D., LÉVY B., PIETRONI N., PUPPO E., SILVA C., TARINI M., ZORIN D.: Quad-mesh generation and processing: A survey. *Comput. Graph. Forum 32*, 6 (2013), 51–76. 3

[BNPB13] BHATIA H., NORGARD G., PASCUCCI V., BREMER P.-T.: The Helmholtz–Hodge decomposition—a survey. *IEEE Transactions on Visualization and Computer Graphics 19*, 8 (2013), 1386–1404. 3

[BZK09] BOMMES D., ZIMMER H., KOBBELT L.: Mixed-integer quadrangulation. *ACM Trans. Graph. 28*, 3 (2009). 3

[CK14] CAMPEN M., KOBBELT L.: Quad layout embedding via aligned parameterization. *Computer Graphics Forum 33*, 8 (2014), 69–81. 3

[CYZL14] CHI M., YAO C., ZHANG E., LEE T.: Optical illusion shape texturing using repeated asymmetric patterns. *The Visual Computer 30*, 6-8 (2014). 2

[DBG*06] DONG S., BREMER P.-T., GARLAND M., PASCUCCI V., HART J. C.: Spectral surface quadrangulation. *ACM Trans. Graph. 25*, 3 (2006), 1057–1066. 2

[dGDT15] DO GOES F., DESBRUN M., TONG Y.: Vector field processing on triangle meshes. In *SIGGRAPH Asia 2015 Courses* (2015), pp. 17:1–17:48. 2

[DHLM05] DESBRUN M., HIRANI A., LEOK M., MARSDEN J.: Discrete exterior calculus. preprint, arXiv:math.DG/0508341, 2005. 3

[DLL*10] DEY T. K., LI K., LUO C., RANJAN P., SAFA I., WANG Y.: Persistent heat signature for pose-oblivious matching of incomplete models. *Computer Graphics Forum 29*, 5 (2010), 1545–1554. 2

[DRW10] DEY T. K., RANJAN P., WANG Y.: Convergence, stability, and discrete approximation of Laplace spectra. In *Symp. Discrete Algorithms* (2010), pp. 650–663. 2

[DVPSH14] DIAMANTI O., VAXMAN A., PANOZZO D., SORKINE-HORNUNG O.: Designing *N*-PolyVector fields with complex polynomials. *Computer Graphics Forum 33*, 5 (2014), 1–11. 3

[DVPSH15] DIAMANTI O., VAXMAN A., PANOZZO D., SORKINE-HORNUNG O.: Integrable PolyVector fields. *ACM Trans. Graph. 34*, 4 (2015). 13

[DWLF12] DE WITT T., LESSIG C., FIUME E.: Fluid simulation using Laplacian eigenfunctions. *ACM Trans. Graph. 31*, 1 (2012), 10:1–10:11. 2

[ECBK14] EBKE H.-C., CAMPEN M., BOMMES D., KOBBELT L.: Level-of-detail quad meshing. *ACM Trans. Graph. 33*, 6 (2014), 184:1–184:11. 3

[FSDH07] FISHER M., SCHRÖDER P., DESBRUN M., HOPPE H.: Design of tangent vector fields. *ACM Trans. Graph. 26*, 3 (2007). 3, 8, 9, 10

[GBAL09] GEBAL K., BÆRENTZEN J. A., AANÆS H., LARSEN R.: Shape analysis using the auto diffusion function. *Computer Graphics Forum 28*, 5 (2009), 1405–1413. 2

[HP11] HILDEBRANDT K., POLTHIER K.: On approximation of the Laplace–Beltrami operator and the Willmore energy of surfaces. *Computer Graphics Forum 30*, 5 (2011), 1513–1520. 2

[HPW06] HILDEBRANDT K., POLTHIER K., WARDETZKY M.: On the convergence of metric and geometric properties of polyhedral surfaces. *Geometricae Dedicata 123* (2006), 89–112. 2

[HSvTP10] HILDEBRANDT K., SCHULZ C., VON TYCOWICZ C., POLTHIER K.: Eigenmodes of surface energies for shape analysis. In *Proceedings of Geometric Modeling and Processing* (2010), pp. 296–314. 2

[HSvTP12] HILDEBRANDT K., SCHULZ C., VON TYCOWICZ C., POLTHIER K.: Modal shape analysis beyond Laplacian. *Computer Aided Geometric Design 29*, 5 (2012), 204–218. 2

[HWAG09] HUANG Q., WICKE M., ADAMS B., GUIBAS L.: Shape decomposition using modal analysis. *Computer Graphics Forum 28*, 2 (2009), 407–416. 2

[HZ00] HERTZMANN A., ZORIN D.: Illustrating smooth surfaces. In *Proc. SIGGRAPH 2000* (2000). 2

[HZM*08] HUANG J., ZHANG M., MA J., LIU X., KOBBELT L., BAO H.: Spectral quadrangulation with orientation and alignment control. *ACM Trans. Graph. 27*, 5 (2008), 1–9. 2

[IBB15] IARUSSI E., BOMMES D., BOUSSEAU A.: Bendfields: Regularized curvature fields from rough concept sketches. *ACM Trans. Graph. 34*, 3 (2015). 2

[KCPS13] KNÖPPEL F., CRANE K., PINKALL U., SCHRÖDER P.: Globally optimal direction fields. *ACM Trans. Graph. 32*, 4 (2013), 59:1–59:10. 3

[KCPS15] KNÖPPEL F., CRANE K., PINKALL U., SCHRÖDER P.: Stripe patterns on surfaces. *ACM Trans. Graph. 34* (2015). 2

[KG00] KARNI Z., GOTSMAN C.: Spectral compression of mesh geometry. In *ACM SIGGRAPH* (2000), pp. 279–286. 2, 11

[KNP07] KÄLBERER F., NIESER M., POLTHIER K.: Quadcover - surface parameterization using branched coverings. *Computer Graphics Forum 26*, 3 (2007). 3

[LHJ*14] LING R., HUANG J., JÜTTLER B., SUN F., BAO H., WANG W.: Spectral quadrangulation with feature curve alignment and element size control. *ACM Trans. Graph. 34*, 1 (2014), 11:1–11:11. 2

[LLW15] LI Y., LIU Y., WANG W.: Planar hexagonal meshing for architecture. *IEEE Trans. Vis. Comput. Graph. 21*, 1 (2015). 3

[LLZ*11] LI E., LÉVY B., ZHANG X., CHE W., DONG W., PAUL J.-C.: Meshless quadrangulation by global parameterization. *Computers & Graphics* (2011). 3

[LMH*15] LIU B., MASON G., HODGSON J., TONG Y., DESBRUN M.: Model-reduced variational fluid simulation. *ACM Trans. Graph. 34*, 6 (2015), 244:1–244:12. 2

[LZ09] LÉVY B., ZHANG H.: Spectral mesh processing. In *ACM SIGGRAPH ASIA Courses* (2009), pp. 1–47. 2

[MDSB03] MEYER M., DESBRUN M., SCHRÖDER P., BARR A. H.: Discrete differential-geometry operators for triangulated 2-manifolds. In *Visualization and Mathematics III*. Springer, 2003, pp. 35–57. 2, 6

[MRMH12] MEHTA S. U., RAMAMOORTHI R., MEYER M., HERY C.: Analytic tangent irradiance environment maps for anisotropic surfaces. *Computer Graphics Forum 31*, 4 (2012). 2

[OBCS*12] OVSJANIKOV M., BEN-CHEN M., SOLOMON J., BUTSCHER A., GUIBAS L.: Functional maps: A flexible representation of maps between shapes. *ACM Trans. Graph. 31*, 4 (2012), 30:1–30:11. 2

[OMMG10] OVSJANIKOV M., MÉRIGOT Q., MÉMOLI F., GUIBAS L.: One point isometric matching with the heat kernel. *Computer Graphics Forum 29*, 5 (2010), 1555–1564. 2

[OSG08] OVSJANIKOV M., SUN J., GUIBAS L.: Global intrinsic symmetries of shapes. *Computer Graphics Forum 27*, 5 (2008), 1341–1348. 2

[PLS*15] PAN H., LIU Y., SHEFFER A., VINING N., LI C.-J., WANG W.: Flow aligned surfacing of curve networks. *ACM Trans. Graph. 34*, 4 (2015). 2

[PP93] PINKALL U., POLTHIER K.: Computing discrete minimal surfaces and their conjugates. *Experimental Mathematics 2*, 1 (1993), 15–36. 2, 5, 6

[PP00] POLTHIER K., PREUSS E.: Variational approach to vector field decomposition. In *Proc. Eurographics Workshop on Scientific Visualization* (2000), pp. 147–156. 3, 4

[PP03] POLTHIER K., PREUSS E.: Identifying vector field singularities using a discrete Hodge decomposition. In *Visualization and Mathematics III* (2003). 3, 4

[RGB*14] RAYMOND B., GUENNEBAUD G., BARLA P., PACANOWSKI R., GRANIER X.: Optimizing brdf orientations for the manipulation of anisotropic highlights. *Comput. Graph. Forum 33*, 2 (2014). 2

[RLL*06] RAY N., LI W. C., LÉVY B., SHEFFER A., ALLIEZ P.: Periodic global parameterization. *ACM Trans. Graph. 25*, 4 (2006), 1460–1485. 3

[ROA*13] RUSTAMOV R. M., OVSJANIKOV M., AZENCOT O., BEN-CHEN M., CHAZAL F., GUIBAS L.: Map-based exploration of intrinsic shape differences and variability. *ACM Trans. Graph. 32*, 4 (2013), 72:1–72:12. 2

[Rus07] RUSTAMOV R. M.: Laplace–Beltrami eigenfunctions for deformation invariant shape representation. In *Symposium on Geometry Processing* (2007), pp. 225–233. 2

[RVLL08] RAY N., VALLET B., LI W. C., LÉVY B.: N-symmetry direction field design. *ACM Trans. Graph. 27*, 2 (2008). 3

[RWP05] REUTER M., WOLTER F.-E., PEINECKE N.: Laplace-spectra as fingerprints for shape matching. In *Symposium on Solid and Physical Modeling* (2005), pp. 101–106. 2

[RWP06] REUTER M., WOLTER F.-E., PEINECKE N.: Laplace–Beltrami spectra as "Shape-DNA" of surfaces and solids. *Computer-Aided Design 38*, 4 (2006), 342–366. 2

[SBCBG11] SOLOMON J., BEN-CHEN M., BUTSCHER A., GUIBAS L.: Discovery of intrinsic primitives on triangle meshes. vol. 30, pp. 365–374. 2

[SHKvL09] SHARMA A., HORAUD R. P., KNOSSOW D., VON LAVANTE E.: Mesh segmentation using Laplacian eigenvectors and Gaussian mixtures. In *Manifold Learning and Its Applications* (2009). 2

[SOG09] SUN J., OVSJANIKOV M., GUIBAS L. J.: A concise and provably informative multi-scale signature based on heat diffusion. *Computer Graphics Forum 28*, 5 (2009), 1383–1392. 2

[TLHD03] TONG Y., LOMBEYDA S., HIRANI A. N., DESBRUN M.: Discrete multiscale vector field decomposition. *ACM Trans. Graph. 22*, 3 (2003), 445–452. 3, 4

[TPP*11] TARINI M., PUPPO E., PANOZZO D., PIETRONI N., CIGNONI P.: Simple quad domains for field aligned mesh parametrization. *ACM Trans. Graph. 30*, 6 (2011). 3

[VCD*16] VAXMAN A., CAMPEN M., DIAMANTI O., PANOZZO D., BOMMES D., HILDEBRANDT K., BEN-CHEN M.: Directional Field Synthesis, Design, and Processing. *Computer Graphics Forum 35*, 2 (2016). 2

[VL08] VALLET B., LÉVY B.: Spectral geometry processing with manifold harmonics. *Computer Graphics Forum* (2008). 2, 7, 11

[VMHB14] VÁŠA L., MARRAS S., HORMANN K., BRUNNETT G.: Compressing dynamic meshes with geometric Laplacians. *Computer Graphics Forum 33*, 2 (2014), 145–154. 2, 11

[War06] WARDETZKY M.: *Discrete Differential Operators on Polyhedral Surfaces–Convergence and Approximation*. PhD thesis, Freie Universität Berlin, 2006. 3, 4

[War13] WARNER F. W.: *Foundations of differentiable manifolds and Lie groups*, vol. 94. Springer, 2013. 3

[WBH*07] WARDETZKY M., BERGOU M., HARMON D., ZORIN D., GRINSPUN E.: Discrete quadratic curvature energies. *Comput. Aided Geom. Des. 24*, 8-9 (2007). 4, 5

[WGH12] WAGNER C., GARTH C., HAGEN H.: Harmonic field analysis. In *New Developments in the Visualization and Processing of Tensor Fields*. Springer, 2012, pp. 363–379. 2

[WL01] WEI L.-Y., LEVOY M.: Texture synthesis over arbitrary manifold surfaces. In *SIGGRAPH* (2001). 2

[WMKG07] WARDETZKY M., MATHUR S., KÄLBERER F., GRINSPUN E.: Discrete Laplace operators: No free lunch. In *Symposium on Geometry Processing* (2007), pp. 33–37. 2

[YCLJ12] YAO C.-Y., CHI M.-T., LEE T.-Y., JU T.: Region-based line field design using harmonic functions. *IEEE Trans. Vis. Comput. Graph. 18*, 6 (2012). 2

[ZHT07] ZHANG E., HAYS J., TURK G.: Interactive tensor field design and visualization on surfaces. *IEEE Trans. Vis. Comput. Graph. 13*, 1 (2007). 3

[ZMT06] ZHANG E., MISCHAIKOW K., TURK G.: Vector field design on surfaces. *ACM Trans. Graph. 25*, 4 (2006), 1294–1326. 3

[ZvKD10] ZHANG H., VAN KAICK O., DYER R.: Spectral mesh processing. *Computer Graphics Forum 29*, 6 (2010), 1865–1894. 2

[ZZCJ14] ZHUANG Y., ZOU M., CARR N., JU T.: Anisotropic geodesics for live-wire mesh segmentation. *Comput. Graph. Forum 33*, 7 (2014). 2