Scale-Invariant Directional Alignment of Surface Parametrizations

Marcel Campen[1]   Moritz Ibing[2]   Hans-Christian Ebke[2]   Denis Zorin[1]   Leif Kobbelt[2]

[1]New York University   [2]RWTH Aachen University

Abstract

Various applications of global surface parametrization benefit from the alignment of parametrization isolines with principal curvature directions. This is particularly true for recent parametrization-based meshing approaches, where this directly translates into a shape-aware edge flow, better approximation quality, and reduced meshing artifacts. Existing methods to influence a parametrization based on principal curvature directions suffer from scale-dependence, which implies the necessity of parameter variation, or try to capture complex directional shape features using simple 1D curves. Especially for non-sharp features, such as chamfers, fillets, blends, and even more for organic variants thereof, these abstractions can be unfit. We present a novel approach which respects and exploits the 2D nature of such directional feature regions, detects them based on coherence and homogeneity properties, and controls the parametrization process accordingly. This approach enables us to provide an intuitive, scale-invariant control parameter to the user. It also allows us to consider non-local aspects like the topology of a feature, enabling further improvements. We demonstrate that, compared to previous approaches, global parametrizations of higher quality can be generated without user intervention.

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

1. Introduction

Global parametrizations of surfaces have been made widespread use of in recent years for the purpose of high-quality remeshing, in particular with semi-structured quad meshes [KNP07, BZK09, MPKZ10, KMZ11, PTSZ11, BCE*13, PPTSH14, CK14b, ECBK14, MPZ14, JTPSH15, CBK15]. It is well-known that aligning the edges of such meshes with the surface's principal curvature directions is beneficial [BLP*13, CK14b, LRL06, ACSD*03]; it can

Figure 1: *Aligning a global parametrization to principal curvature directions everywhere is not reasonably possible in general – it often implies severe distortion. Only certain regions should thus be chosen to enforce alignment in. Colored here are the important, alignable regions automatically selected by our method.*

improve approximation quality [D'A00], reduce aliasing [BK01], and enhance planarity [LXW*11]. Besides these technical reasons, aesthetics are a common additional motivation. Therefore, in the context of parametrization-based meshing, there is a strong need for alignment of parametrization isolines (which induce the mesh edges) with principal curvature directions.

On the other hand, it is also known that enforcing strict alignment (wherever the curvature directions are well defined) can imply arbitrarily large length distortions and even degeneracies or inversions. In the context of quad-*dominant* meshing, this is usually dealt with by introducing non-quad elements [ACSD*03, MK04, JTPSH15] that effectively absorb the changes in vertex density. While a similar effect can be achieved in quad-only meshing through the introduction of clusters of singularities (or T-junctions), this can be contrary to the goal of yielding a highly regular mesh.

Weighted Alignment. A commonly chosen approach therefore is to strive for alignment of a parametrization to principal directions only through *soft* constraints (i.e. objective terms rather than hard constraints), possibly using varying weights, or using (soft or hard) *sparse* constraints, restricted to certain prominent surface regions. We can subsume both strategies under the notion of *weighted alignment*, with a weight field that varies over the surface and may also assume values 0 and ∞ (the latter denoting a hard constraint).

A number of methods have been described in the literature that effectively design such weight fields – though this is not always explicitly phrased in this form. They decide whether and where the

weight field should vanish, where it should be infinite, and how it should behave otherwise. The use of these methods in practical meshing applications revealed a number of significant shortcomings, in particular related to undesirable scale dependence, incoherence, and over-abstraction.

Contribution. We present a novel algorithm for the construction of a weight field that is

- scale invariant,
- coherent,
- non-abstractive,
- non-local.

We discuss these properties in detail in Section 1.1. This combination of desirable properties is not offered by any previous approach. We demonstrate the beneficial effect on parametrization quality in Section 5. Due to the scale invariance of our approach, we are able to expose an intuitive parameter to the user. It can be understood as a generalization of the *dihedral angle threshold*, commonly employed for the detection of sharp feature edges, to the case of smooth directional features such as chamfers, fillets, blends, and, even more importantly, organic freeform variants thereof.

1.1. Desired Properties

We identified the following four properties of a weighting scheme to be important for use in parametrization alignment.

Scale Invariance. The weights computed by almost all available methods are scale dependent. Directional features of a surface might thus receive very different weights if they are of a different scale, even though they are *similar* in shape and effectively look the same to a user; the weights might even change when the model is simply rescaled globally. This turns out to be a problem in practice, especially on organic models, because often some obvious, significant features receive a very low (or zero) weight, while some "false positives" receive a high weight, just because they happen to be on different scales (cf. Figure 2). In methods that can be controlled via a parameter, this behavior could potentially be ameliorated by varying the parameter(s) according to the local surface shape, but this turns out to be a major burden in practice. Some form of automatic adaptation mechanism based on scale-space techniques [PKG03] might be possible, but has not been demonstrated for this specific purpose yet.

We overcome these issues by proposing a weighting scheme that is *scale invariant*. In the schematic on the right this means that the three smooth features (blue) are treated equally by our approach, based on the fact that they all correspond to a, in this case, 90° bending of the surface. The fact that they differ in relative scale, and the question of what absolute scale they have, which with other methods leads to the abovementioned issues, does not play a role for our scale-invariant approach. In fact, a *binary* weight field
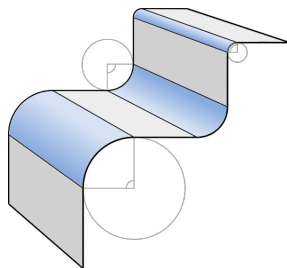
is built, which only distinguishes between regions important and reasonable for alignment, and regions unimportant for alignment.

Coherence. Many previous approaches are local: the weight at a point (usually a vertex or a face) is computed from local properties such as the principal curvatures κ_{max} and κ_{min} at that point. We observe that often isolated or scattered points receive a high weight, as well as points very close but with largely differing principal directions (cf. Figure 3). Especially this lack of directional coherence between neighboring points of high weight can be problematic in applications. In the context of quad remeshing it can, for instance, induce an excessive amount of undesirable irregular vertices (cf. Figures 11 and 10). Our approach takes spatial relationships into account, so as to design an inherently coherent weighting.

Non-abstraction. The design of a *binary* weight field can be phrased as a feature detection problem: set the weight to 1 (or any constant $w > 0$) where the surface contains some kind of directional feature, set it to 0 everywhere else. In this context of directional feature detection, several methods try to abstract from the potentially complex geometric nature of non-sharp feature regions and represent them using one-dimensional curves. While a curve-based representation is clearly suitable for sharp features, it can be quite problematic for soft features. For relatively simple, man-made geometries (with chamfers, fillets, rolling-ball blends), a precise definition of representative curves may exist, but for more organic objects, ambiguities are unavoidable. The common consequence are unintuitive, low-quality results (cf. Figure 4). We respect and exploit the two-dimensional nature of such smooth directional shape features. This not only avoids the abovementioned issues, it is the key to achieving scale invariance and coherence.

Non-locality. Considering the blue and the green regions depicted on the right, one can argue that aligning to principal curvature directions in the blue region is more important than alignment within the very narrow green region. Locally, however, the surface is indistinguishable at points in these two regions (with identical curvature radius). Our method takes non-local measures into account, allowing to distinguish these regions. This non-locality furthermore allows us to take topological aspects of such regions into account for further improvements.
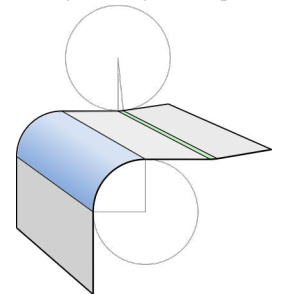
2. Related Work

2.1. Aligned Parametrization

Alignment to principal curvature (or other) directions has been considered in a variety of global parametrization approaches. The alignment terms or constraints can directly be integrated into the parametrization objective itself; this is usually done to achieve strict alignment with sharp feature edges or boundaries [BZK09, MPKZ10, BCE*13]. Another common, more indirect way of infusing directional information into the parametrization process is via a *guiding field*. So-called cross or frame fields are commonly
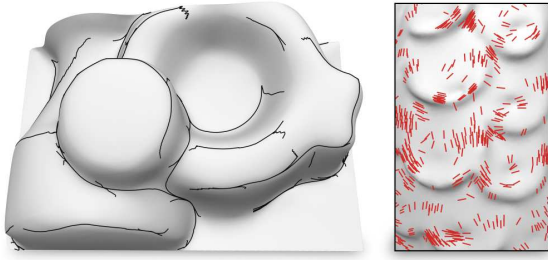
Figure 2: *Scale-dependent threshold parameters easily lead to unintuitive results. For a chosen global threshold, some obvious features may already be missing, while false positives are still present (left: [WG09], right: [BZK09]).*
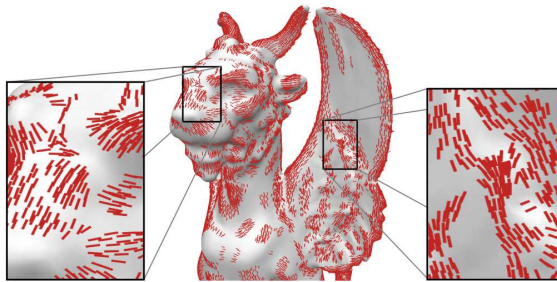

Figure 3: *Pointwise feature detection, in this case [NPPZ12], easily leads to noisy and contradictory results because no notion of coherence is involved.*

Figure 4: *In contrast to sharp features, smooth features cannot always be well-represented by curves. Consequently, methods that try to do so, in this case [WG09], sometimes generate unexpected results. A regional representation is more appropriate in such cases (compare, e.g., the fingers with Figure 1).*

Figure 5: *Several methods are restricted to feature curve networks (black). Fading smooth features (and cyclic features) are neglected (left: [NSP10], right: ours, which can handle fading and cyclic features).*

used to A) determine a suitable set of parametrization singularities or cones which keep the overall distortion at a low level, and B) directionally influence the parametrization by optimizing its isolines for alignment with the field directions via a Poisson formulation [RLL*06, KNP07, BZK09]. Most of the methods for the construction of such guiding fields offer possibilities to (hard or soft) align the field to prescribed directions [VCD*16], which then carries over to an indirect alignment objective for the parametrization.

2.2. Binary Weight Fields

Many previously proposed methods for the computation of a weight field compute a binary weighting, which assigns to every surface point either a value of 0 or a constant $w > 0$. Feature detection approaches implicitly fall into this category: a point is assigned 0 iff it is not considered part of a feature. As we are interested in directional alignment to principal curvature directions, only detectors for directional, edge-like feature are of interest, not detectors for corner-like point features such as variants of SIFT [DK12] or point-centric shape classifiers [MPS*04].

In the special case of *sharp* features, where infinite curvature is concentrated on a curve on the surface, the situation is simple, detection nearly trivial. Only discretization (by means of a triangulation) introduces a degree of complication and requires the specification of, e.g., a threshold on the dihedral angle.

For *smooth* features, however, the situation gets much more complicated. In this case, ambiguities are not only due to discretization, and many different definitions of the notion of smooth directional
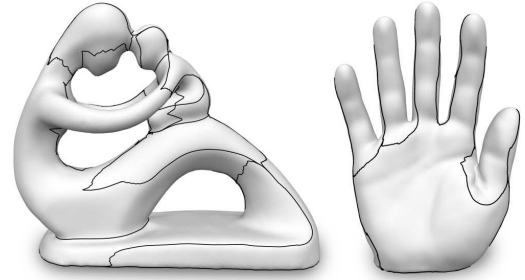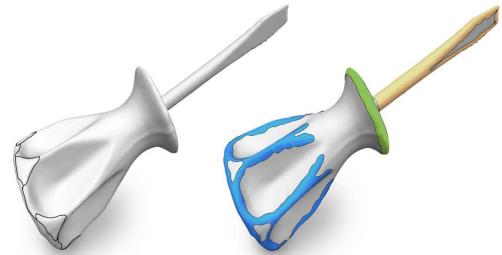
feature are possible. This led to the development of a variety of feature detectors.

Pointwise Detectors. Pointwise feature detection, based on principal curvatures, has been used to determine a binary $\{0, \infty\}$-field to align guiding fields [BZK09, NPPZ12]. In both works, a *shape anisotropy* threshold and a *flatness* threshold are used combinedly. The threshold on shape anisotropy, measured in terms of the principal curvatures, is meant to discriminate regions which are close to cylindrical (i.e. where principal curvatures are significantly distinct in magnitude) from regions which are closer to spherical or hyperbolic. The threshold on flatness is meant to discriminate regions with high from regions with low curvature.

As can be seen in Figures 2 and 3, often isolated or scattered points are detected as features, as well as points very close but with largely differing principal directions. The inherent scale dependence of the thresholds likewise is an issue.

Curve based Detectors. So-called *ridge* and *valley* lines can be detected based on third- and fourth-order differential surface properties [Thi96, OBS04, YBS05, HPW05]. These lines can then be filtered to obtain so-called *crest lines*, representing certain surface feature curves, which in turn can be filtered to discard non-salient or noisy curves. This is generally performed based on integral measures of curvedness or extremality over the individual crest lines. Another class of methods is based on skeletonization. First, regions where the maximum curvature exceeds a threshold are labelled as "curved". These regions are then thinned to a

curve skeleton representing the feature curve structure of the surface [RKS00, KVSM11, NSP10]. Also, a dual strategy can be followed: patches are grown on the surface, and the boundaries where these patches meet are defined to be feature curves [NSP10]. This is a common approach for the feature-sensitive segmentation of surfaces [WHL*13]. Finally, there are topological approaches based on Morse-Smale theory [SWPL08, WG09]. The separatrices of the Morse-Smale complex of a curvature-based scalar field on the surface are considered potential feature curves. They are filtered using topological simplifications based on persistence measures [WG09].

In all cases, the resulting feature curves usually are non-smooth, mainly due to discretization. Some form of post-processing can be necessary to smooth the curves [HPW05, NSP10], which introduces additional parameters and can be a source of robustness issues.

In the context of aligned parametrization one can make use of such curves by setting the weight field to $w > 0$ for the vertices or faces crossed by these curves, or, more directly, by explicitly aligning an isoline of the parametrization with such a curve [CK14a], analogous to the way sharp feature edges are taken into account. However, it can be observed that the abstraction to curves can be problematic in principle (cf. Figure 4), and the scale dependence of most of the threshold or filter parameters likewise is an issue (cf. Figure 2). The methods based on skeletonization are furthermore restricted to closed curve *networks*. Smooth features that fade into a non-feature region are thus generally ignored (cf. Figure 5).

Region based Detectors. There are a few methods which classify regions of a surface according to some predefined set of primitive shapes, for instance based on RANSAC [SWK07], slippage analysis [GG04], or segmentation [WK05]. These are targeted at the detection of, e.g., cylindrical or spherical parts of a surface; applicability to feature detection in a more general sense is limited, in particular on organic shapes.

2.3. Continuous Weight Fields

In the context of cross field alignment, some non-binary weight fields with continuous values, typically based on local principal curvature values κ_{max} and κ_{min}, have been proposed: $|\kappa_{max} - \kappa_{min}| \cdot (\kappa_{max}^2 + \kappa_{min}^2)^{-\frac{1}{2}}$ [KNP07], $|\kappa_{max}/\kappa_{min}|$ [RLL*06], or $(\kappa_{max} - \kappa_{min})^2$ [KCPS13]. One recent method [JTPSH15] uses an objective function for alignment that is non-linear in the deviation from the principal curvature directions (of the elementary edge curvature tensors [ACSD*03, CSM03]), i.e. the corresponding weights cannot be expressed simply as a field of scalars.

Notice that the former two are mostly scale invariant (although especially the second requires special treatment when the denominator vanishes). The purely local nature of these approaches makes the weights agnostic about the difference between the blue and green region depicted in Section 1.1; for instance, the first method assigns a weight of 1 to both of them. In fact, any point where $\kappa_{min} = 0$ and $\kappa_{max} = \varepsilon > 0$ receives a weight of 1. This is a major problem in practice, because points in (nearly) planar regions, where the principal directions are effectively random, receive a high or even maximum weight, just due to numerical inaccuracies. Note that this is not a specific issue of this one measure; for *any* measure that is scale invariant and local, this is necessarily the case.

This is the reason why some more recent methods discussed earlier, [BZK09, NPPZ12], add an additional planarity cut-off parameter – which, however, breaks the scale invariance. In our method this problem is solved instead by the use of non-local measures.

2.4. Principal Curvatures

A number of methods for the estimation of principal curvatures and principal curvature direction on discrete surfaces have been described, e.g. [CSM03, ACSD*03, MOG09, HPW05]. Most of the discussed methods for weight field generation, just like our method, are complementary to this process, i.e. the curvature values and directions are taken as input with no regard to their origin. Dealing with discretization artifacts or data noise is due to the principal curvature estimator in this context – most of the discussed papers, like ours, do not make a contribution in this regard, the provided curvature values and directions are taken as is.

3. Smooth Regions

Let $\kappa_{min}(x)$ and $\kappa_{max}(x)$ be the principal curvatures on a surface, with $|\kappa_{max}(x)| \geq |\kappa_{min}(x)|$, i.e. κ_{max} refers to the principal curvature with largest *absolute* value. The unit tangent vector fields $\mathbf{a}_{min}(x)$ and $\mathbf{a}_{max}(x)$ with $\mathbf{a}_{min} = \mathbf{a}_{max}^{\perp}$ are the corresponding principal curvature direction fields, with vector orientation chosen consistently in a neighborhood of a non-umbilical point x. By γ_{min} and γ_{max} we denote lines of curvature, i.e. integral curves of $\mathbf{a}_{min}, \mathbf{a}_{max}$.

One can measure the rate of change $\kappa_{\mathbf{v}}^{g}$ of a unit vector field $\mathbf{a}$ at a point p in unit tangent direction $\mathbf{v}$ using the norm of the covariant derivative: $\kappa_{\mathbf{v}}^{g} = \|\nabla_{\mathbf{v}}\mathbf{a}\|$ [dC76, §4.4, §5.10]. Note that in directions $\mathbf{v} = \mathbf{a}_{min}, \mathbf{a}_{max}$ it corresponds to the *tangential* or *geodesic curvature* of the lines of curvature; we generally call $\kappa_{\mathbf{v}}^{g}$ the *geodesic curvature* in direction $\mathbf{v}$. Note that due to $\mathbf{a}_{min} = \mathbf{a}_{max}^{\perp}$ (and the fact that the Levi-Civita connection ∇ inducing the covariant derivative is a metric connection [Spi79, §6]) the rate of change in a given direction $\mathbf{v}$ is the same for both fields, i.e. we have $\|\nabla_{\mathbf{v}}\mathbf{a}_{min}\| = \|\nabla_{\mathbf{v}}\mathbf{a}_{max}\|$. The *maximum geodesic* curvature κ^{g} at a point p is defined as the maximum of the geodesic curvature $\kappa_{\mathbf{v}}^{g}$ over all tangent directions $\mathbf{v}$ at p:

$$\kappa^{g} = \max_{\|\mathbf{v}\|=1} \|\nabla_{\mathbf{v}}\mathbf{a}_{min}\| = \max_{\|\mathbf{v}\|=1} \|\nabla_{\mathbf{v}}\mathbf{a}_{max}\|. \quad (1)$$

At umbilical points, where the principal directions are not well-defined, we can assume $\kappa^{g} = \infty$ for our purposes.

Informally, maximum geodesic curvature measures how much the network of curvature lines near a point deviates from a grid of straight geodesic lines. For example, it is exactly zero for a cylinder.

3.1. Definition

We define a *smooth region* as a connected component of the surface where (for coherence) the maximum geodesic curvature is (in a scale invariant sense) low, as defined more precisely below. Using an additional definition of a *region significance*, which involves a user-adjustable threshold, we identify (in a non-local manner) a subset of the smooth regions, corresponding to regions deemed important and suitable for alignment.

Smooth Regions   If we would require κ^g to be 0, only perfectly linear smooth features, e.g. regions of cylindrical shape, would be captured as smooth regions. As one is also interested in non-linear "bent" regions, e.g. of toroidal shape, regions with $\kappa^g > 0$ must be considered, too. To achieve scale-invariant behavior, we consider κ^g in relation to $|\kappa_{\max}|$, i.e. *tangential* curvature in relation to *normal* curvature. The inset shows a prototypical situation (the blue region on a toroidal surface, with constant $\kappa_{\max}$ as indicated by the osculating circles) under increasing degree of "bending" κ^g

$\kappa^g = \frac{1}{30}|\kappa_{\max}|$

$\kappa^g = \frac{1}{3}|\kappa_{\max}|$

$\kappa^g = |\kappa_{\max}|$

$\kappa^g = 3|\kappa_{\max}|$

(as indicated by circular arcs). The point where the torus collapses to a topological sphere is where κ^g matches $|\kappa_{\max}|$. We define a smooth region as a maximal connected region S with

$$\kappa^g(x) < |\kappa_{\max}(x)|, \text{ for all } x \in S. \tag{2}$$

This natural criterion turns out to perform so well in general that an additional (adjustable) factor proved to be superfluous.

Significance   As outlined above, we deem particularly important that the significance measure is scale-invariant. Therefore, we consider how much the surface "bends across the region" – independent of how smooth or sharp its shape is. Our significance value is the angle through which the surface normal maximally varies along any streamline of $\mathbf{a}_{\max}$ (red) across the smooth region ($90°$ in the inset). More specifically, consider all curvature lines $\gamma_{\max}$ on a region, each parametrized by a unit-speed parameter s. We define the significance of a smooth region $\mathcal{F}$ as

$$\angle(\mathcal{F}) = \max_{\gamma_{\max}} \max_{s_0 < s_1} \left| \int_{s_0}^{s_1} \kappa_{\max}(\gamma_{\max}(s))\, ds \right|. \tag{3}$$

The integral measures the deviation between normals at s_0 and s_1 on the curvature line $\gamma_{\max}$. This notion of significance can be seen as a generalization of the intuitive dihedral angle measure often used for sharp feature detection. Some average over all the $\gamma_{\max}$ could be chosen as well, but we found taking the maximum to behave more intuitively, in particular on fading features.

A threshold on $\angle(\mathcal{F})$ is offered as filter parameter by our method. Figure 6 illustrates the effect of it. In all following experiments, we used a setting of $70°$; model-dependent tuning is not necessary mainly due to scale invariance. Figure 1 shows the filtered smooth regions on a number of models.

Note that the smooth regions, filtered by significance, fulfill the four properties initially set forth: 1) they are defined scale invariantly, 2) the field of principal directions is coherent (bounded κ^g) within the regions by definition, 3) the significance measure is non-local, and 4) we do not abstract regions to mere curves.

3.2. Discretization & Implementation

Principal Curvature   In our implementation, we use the technique described by Alliez et al. [ACSD*03] to estimate curvature tensors from which principal curvature directions $\mathbf{a}_{\min}$ and $\mathbf{a}_{\max}$, normals, and magnitudes $\kappa_{\min}$ and $\kappa_{\max}$ can be extracted through eigendecomposition. The normal and $\mathbf{a}_{\max}$ can, however, be unstable where their eigenvalues are close. We thus estimate surface normals $\mathbf{n}$ separately: in line with the curvature tensor integration, face normals are averaged to obtain $\mathbf{n}'$, which is then made orthogonal to $\mathbf{a}_{\min}$ via $\mathbf{n} = \mathbf{a}_{\min} \times \mathbf{n}' \times \mathbf{a}_{\min}$. Finally, $\mathbf{a}_{\max}$ is determined as the direction orthogonal to $\mathbf{a}_{\min}$ and $\mathbf{n}$.

For the purpose of representation and computation, we consider the line fields $\mathbf{a}_{\min}$ and $\mathbf{a}_{\max}$ as unit vector fields, with one vector per face each. Note that these fields are not tangent in general. Where we need these directions to lie in the supporting planes of the triangles (specifically: when tracing streamlines and when setting up constraints) we use orthogonal projections onto the planes.

Geodesic Curvature   We approximate the curvature κ^g per triangle. For each triangle, we take the maximum over a directional (covariant) derivative across each of its edges. We evaluate these directional derivatives using finite differences on the barycentric dual mesh. For an edge e, with d being the distance of the two adjacent faces' f_1 and f_2 barycenters, the discrete directional derivative of $\mathbf{a}_{\min}$ across e is easily calculated as $\kappa(e) = \min\|\mathbf{a}_{\min}(f_1) \pm \mathbf{a}_{\min}(f_2)\|/d$ (where the sign invariance accounts for the arbitrarily chosen orientation of the $\mathbf{a}_{\min}$ vectors). Note that the tangent vectors $\mathbf{a}_{\min}$ living in different tangent spaces first need to be transformed into a common one using the Levi-Civita connection. In the discrete case, this amounts to a simple rotation about the axis $\mathbf{n}_1 \times \mathbf{n}_2$ by $\arccos(\mathbf{n}_1 \cdot \mathbf{n}_2)$.

If this value $\kappa(e)$ exceeds the threshold ($|\kappa_{\max}(f)|$) for any one of the three adjacent edges, the triangle f is not part of a smooth region ("non-smooth"); otherwise, it is ("smooth"). The smooth regions are the connected components of the set of smooth triangles.

Note that a smooth region can be arbitrarily large or small, even consist of a single triangle. Such small regions may appear in near-umbilic regions, where the principal directions are basically just noise, as shown here on a discretized sphere ($\mathbf{a}_{\max}$ red, smooth regions blue). Such regions are naturally discarded by the significance threshold (here the significance of each one is less than $1°$).

Significance   We approximate $\angle(\mathcal{F})$ in (3) based on a finite sample of curvature lines $\gamma_{\max}$: starting from each vertex on the boundary of a smooth region, a streamline of $\mathbf{a}_{\max}$ is traced by following the piecewise-constant field $\mathbf{a}_{\max}$ face-by-face until a boundary is reached again.

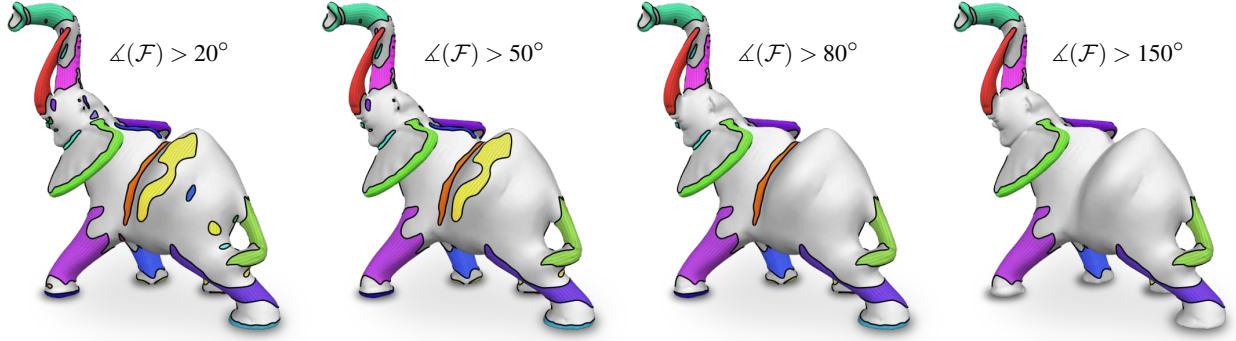For each one of these streamlines, the inner maximum of (3)
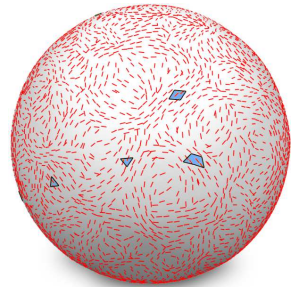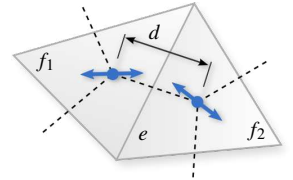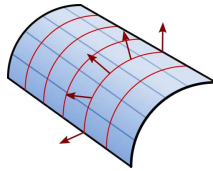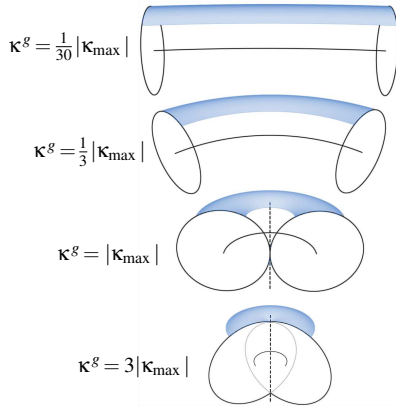
Figure 6: *Effect of the significance angle threshold on $\measuredangle(\mathcal{F})$ used to filter smooth regions in a non-local, scale invariant manner.*

needs to be evaluated. Based on the equivalence

$$\max_{s_0 < s_1} \left| \int_{s_0}^{s_1} g \right| = \max_{s_0, s_1} \left| \int_{s_0}^{s_1} g \right| = \max_{s_0, s_1} \left| \int_0^{s_1} g - \int_0^{s_0} g \right| = \max_s \int_0^s g - \min_s \int_0^s g,$$

this is easily done as follows. Let f_i, $i \in \{0, \ldots, n\}$ be the sequence of faces crossed by a streamline γ_{max}. The signed normal change from f_i to f_{i+1} is $\beta_i = \pm \arccos(\mathbf{n}_i \cdot \mathbf{n}_{i+1})$. With $\int_0^s \kappa_{max}(\gamma_{max}(s)) ds \approx \sum_{i=0}^{j(s)} \beta_i$ the inner maximum is computed as $\measuredangle(\gamma_{max}) = \max_j \sum_{i=0}^j \beta_i - \min_j \sum_{i=0}^j \beta_i$.

Topology   This way of computing $\measuredangle(\mathcal{F})$ is appropriate for smooth regions with disc topology. Note, however, that also topologically more complex regions can arise, e.g. around cylindrical shapes like extremities of character models. Such *cyclic regions* can contain cyclic (or nearly cyclic, helical) streamlines. The same is possible if the $\mathbf{a}_{max}$ field contains a singularity within a region (which could occur at the tip of pointed, needle-like shapes). As the above significance angle computation technique does not capture cyclic streamlines (since they do not intersect the region's boundary), we detect such configurations in advance in a conservative manner and assign $\measuredangle(\mathcal{F}) = 360°$:
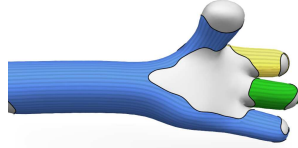
The field's turning number along a cyclic (or short-cut helical) streamline is necessarily zero. Thus only around a region boundary loop (or singular vertex) with turning number zero (or, theoretically, a set of boundaries and singularities with a combined turning number of zero) a cyclic or near-cyclic streamline can be contained in a region. The inset shows an illustration of region (blue) boundaries (black) with turning number zero (top) and one (bottom). We thus compute the turning number [RVLL08, CDS10, PZ07] of the $\mathbf{a}_{max}$ field around every boundary loop and every vertex of a region and consider a region cyclic if one or more of these turning numbers is zero.
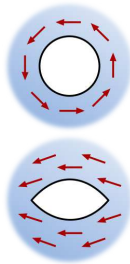
Run Time   Note that none of the steps in the determination of the smooth regions and the computation of the significance angles is computationally intensive. For instance, on a large mesh with 500K triangles and ~ 2000 smooth regions, our implementation runs for about 1s.

4. Parametrization Alignment

In this section, we show how the filtered smooth regions can be used to create weight fields for parametrization alignment to principal curvature directions, and how the specific properties of our method provide additional advantages in this context.

4.1. Cross Field Constraints

Most obviously, smooth principal curvature directions inside the filtered smooth regions can be directly used to directionally hard-constrain the optimization of global direction (cross) fields [BZK09, RVAL09, DVPSH14]. This means we use a $\{0, \infty\}$-weight field, with a value of ∞ in the filtered smooth regions. Due to directional coherence, otherwise common artifacts like high distortion or implausible singularities are generally not caused by these hard constraints. Using algorithms for field-guided parametrization (cf. Section 2.1), these aligned direction fields can be exploited to generate parametrizations $P = (u, v)$ whose u- and v-isolines (indirectly) have a tendency for alignment to principal curvature directions in the detected smooth regions. This is achieved by optimizing the parametrization's isolines for being parallel to the guiding field directions (i.e. its gradients for being orthogonal to them).

4.2. Parametrization Constraints

When using the method described in Section 4.1, the guiding field exactly follows the principal directions in the filtered smooth regions, but the parametrization's isolines only follow the guiding field as well as possible (in a least squares sense). Effectively, a compromise between isometry and alignment of the parametrization is made in this way, and it is made in a globally uniform way.

One may argue that in the filtered smooth regions alignment is more important. However, strict alignment of the parametrization to principal directions using hard-constraints (like typically done for sharp features [BZK09, MZ13]) throughout larger regions (in contrast to just along infinitesimal sharp features) is problematic, or even impossible in the discrete setting:

To achieve strict alignment one would need to constrain $\nabla u = \lambda \mathbf{a}_{max}$ (or $\nabla v = \lambda \mathbf{a}_{max}$, depending on the local parametric orientation) within the filtered smooth regions to make the parametrization's gradient coincide with $\mathbf{a}_{max}$ (thus its isolines align with $\mathbf{a}_{min}$). Here the vectors $\mathbf{a}_{max}$ are assumed to be oriented consistently (up to a cut graph where they, like the parametrization, undergo

transitions [KNP07, BZK09]). $\lambda > 0$ is a free, variable scalar field. This constraint system can be infeasible for non-simply connected regions, or even for simply-connected ones in the discrete piecewise linear setting. Allowing non-positive λ guarantees feasibility, but those imply parametrizations with degeneracies and inversions.

As a workaround, one could strive to compute an approximation of $\mathbf{a}_{max}$ which *can* be aligned to with $\lambda > 0$, e.g. through Hodge decomposition, but the result might contain arbitrarily large parametric distortions – additional measures to find a compromise would be necessary. We thus favour the following, more direct approach:

In addition to the $\{0, \infty\}$-weight field used for the guiding cross field optimization, we employ a $\{0, w\}$-weight field in the parametrization computation (with a constant w), adjusting the isometry/alignment balance towards a stronger alignment objective within the filtered smooth regions. Let unit vectors $\mathbf{u}_i$ and $\mathbf{v}_i$ represent the two orthogonal directions of the cross field in face i. Assume, w.l.o.g., cross field direction $\mathbf{v}$ is always the one aligned with the line field $\mathbf{a}_{min}$ of minimum principal curvature direction in the filtered smooth regions, $\mathbf{u}$ is orthogonal to $\mathbf{a}_{min}$. A_i is the area of face i. We take the usual Poisson term [BZK09]

$$E_{\text{Poisson}}(u, v) = \sum_i A_i(\|\nabla u - \mathbf{u}_i\|^2 + \|\nabla v - \mathbf{v}_i\|^2) \to \min,$$

and add to it the alignment term

$$E_{\text{align}}(u) = \sum_i w_i A_i (\nabla u \cdot \mathbf{a}_{min})^2 \to \min,$$

controlled by the binary weight field w_i. Note that, as in the case of sharp feature curves, alignment of isolines to the direction of *minimum* absolute curvature, i.e. along the feature, is the crucial aspect. The alignment term thus propagates orthogonality of the gradient ∇u to $\mathbf{a}_{min}$, i.e. alignment of u-isolines with $\mathbf{a}_{min}$.

We point out that, as the cross field is aligned in the strip regions ($\mathbf{v}_i = \mathbf{u}_i^\perp = \mathbf{a}_{min}$), the combination of both terms is equivalent to

$$E_{\text{Poisson}}(u, v) + E_{\text{align}}(u) = \sum_i A_i(\|\nabla u - \mathbf{u}_i\|^2_{1, w_i+1} + \|\nabla v - \mathbf{v}_i\|^2),$$

where $\|\nabla u - \mathbf{u}\|^2_{\alpha, \beta} = \alpha((\nabla u - \mathbf{u}) \cdot \mathbf{u})^2 + \beta((\nabla u - \mathbf{u}) \cdot \mathbf{u}^\perp)^2 = \alpha(\nabla u \cdot \mathbf{u} - 1)^2 + \beta(\nabla u \cdot \mathbf{u}^\perp)^2$ is an anisotropic norm (relative to frame $(\mathbf{u}, \mathbf{u}^\perp)$). A norm of this kind was used by Bommes et al. [BZK09] globally. In our case it comes into effect locally and only with respect to the direction of minimum curvature.
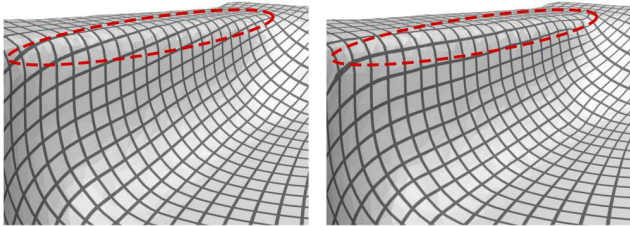

Figure 7: *Left: alignment to soft features can be poor with the traditional approach of constraining a cross field which then guides the parametrization. Right: with our dual weight field approach, better alignment is achieved.*

We observe a factor of 5 to be a good general setting, i.e. we use a $\{0, 5\}$-weight field for the parametrization (the w_i in the above terms), in addition to the $\{0, \infty\}$-weight field used for the guiding cross field (cf. Section 4.1).

Figure 7 demonstrates the alignment improvement achieved by this dual strategy of using a weight field for the guiding field as well as for the parametrization, relative to the results obtained by only explicitly aligning the guiding field followed by a global least-squares fit of the parametrization gradients to the guiding field.

4.3. Topology Constraints

In addition to the *geometrical* principal direction constraints, we can automatically derive *topological* constraints for the parametrization from the filtered smooth regions. For cyclic smooth regions (cf. Section 3.2), we can require that the parametrization be cyclic as well – in the sense that isolines form closed loops in these regions. This automatically prevents so-called *helices*, which have been identified as the structurally most problematic configurations in quad meshes [BLK11], from occurring across the cyclic regions.

Technically these topological constraints are implemented in analogy to the cone alignment constraints described in [MPKZ10]: a path's end point is constrained to get the same u (or v) coordinate as the path's start point, taking the transition functions of the global parametrization along the path into account. In our case, the paths are closed loops, i.e. start and end point are identical. We set up such a constraint for each smooth region boundary loop with turning number zero. It is important to note that this loop's geometry, thus the shape of the smooth region's boundary has no effect – it is only the loop's homotopy class that matters for this *topological* constraint. In particular does this type of constraint not fix an isoline to a prescribed curve (in contrast to *geometric* loop alignment constraints [CK14a]), it only enforces the existence of cyclic isolines. Figure 8 demonstrates the effect of these constraints. Previous weight field computation approaches (cf. Section 2) provide no immediate means to address this topological aspect.

5. Comparison

We already discussed a variety of qualitative arguments for our approach (scale-invariant, coherent, intuitive, no curve abstraction issues) and illustrated these advantages on several examples.
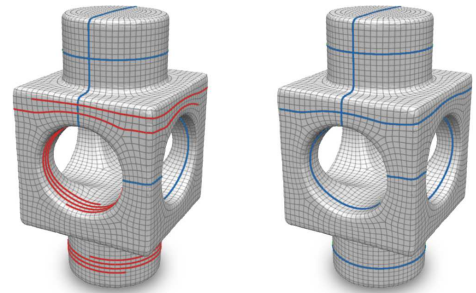

Figure 8: *Left: parametrization result without, right: with topological constraints. Some isolines are highlighted: simple cycles (blue) and helices (red). The problematic helical isolines are automatically avoided by the constraints derived from cyclic smooth regions. Note that they do not necessarily prevent all such helices.*

Performing a quantitative comparison of the weight field generation (and related alignment) approaches is not an easy task. There is no generally accepted quality metric – in particular not a generic, application-agnostic one. What we can do is assess the impact of using different methods based on various properties of the resulting parametrization. In detail, we use the guiding field and parametrization optimization technique of [BZK09] (noting that recent developments [BCE*13, ECBK14, PPTSH14, DVPSH15] relying on the same basis could also be used) and measure the following:

- **stretch**: average (unsigned) relative magnitude deviation of the u- and v-gradients from unity.
- **shear**: average deviation of the angle between u- and v-gradients from $90°$.
- **curvature**: average geodesic curvature of the u- and v-gradient fields.
- **misalignment**: avg. weighted deviation from the principal directions, $(\kappa_{min} - \kappa_{max})^2$-weighted as in [KCPS13, CK14b].
- **singularities**: number of guiding field singularities (i.e. parametrization cones).

We compare against the pointwise binary weigh field generation techniques [BZK09, NPPZ12] outlined in Section 2.2 – the curve-based ones generally proved too problematic for the purpose of constraining the parametrization due to the issues (e.g. jaggedness) discussed in Section 2. Let us stress that here we are not referring to the parametrization optimization techniques presented in these papers [BZK09, NPPZ12], but specifically to the feature discriminators they propose. For comparability, we do not perform manual individual parameter tuning but use the default settings recommended in the respective papers, as well as our default angle threshold of $70°$. For plain comparison of the weight field quality, the topological constraints have not been used here. The same input (the fields $\mathbf{a}_{max}$, $\mathbf{a}_{min}$, κ_{max}, κ_{min}) has been used for each method.

Figure 9 shows the obtained values, averaged over 20 example models (among others, all the models shown throughout the paper). We see that, except for very small differences in stretch, lower values of shear, curvature, and misalignment are achieved by our approach, at the same time yielding a lower number of singularities. Figure 11 shows a close-up comparison on one of the models, demonstrating the typical differences observed: rather large features are often missed by [NPPZ12], whereas [BZK09] filters less aggressively by default, often not discarding spurious and noisy constraints, leading to distortions and excess singularities.

Figure 10 shows the resulting parametrizations on further models. Notice that large scale features (e.g. in the middle column: the foreleg of the elephant or the brim of the vase) are often ignored by the scale-dependent detectors, leading to major misalignment. The generally much larger number of singularities (red and blue dots in Figure 10) and occurrence of singularity clusters in the middle [NPPZ12] and right [BZK09] column can be explained by the fact that small, high frequency details (large κ_{max}) are considered to be the most important features by scale-dependent methods. We emphasize that, while a higher number of singularities can sometimes be advantageous [MZ13], these additional singularities are truly spurious artifacts here: in Figure 12 we visualize the parametric distortion of the same examples and observe that these additional singularities rather reduce than improve the quality.
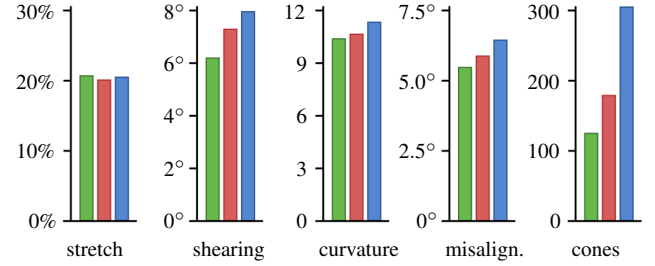
Figure 9: *Comparison of various parametrization quality aspects depending on the method that is employed for weight field computation. ■ Our Smooth Regions, ■ [NPPZ12], ■ [BZK09]. With our method, all properties but stretch are improved: lower shearing, lower parametrization curvature, better principal direction alignment, lower number of singularities (cones).*

6. Limitations & Future Work

Multiple, semantically distinct smooth regions may be captured by one common region if they happen to be adjacent or in close proximity. It would be advantageous if they could automatically be separated, such that each one of them is assigned its own significance value and is filtered accordingly. This will require the consideration of some kind of higher level structural information.

Our construction ensures intra-region coherence of principal directions. The difference of principal directions in different smooth regions in close proximity, however, is not bounded, i.e. there is no guaranteed inter-region coherence. The benefit of some kind of region pruning approach which removes inter-region incoherent parts could be investigated in future research, though our experiments did not reveal undesirable effects related to this matter.

In surface regions with mean curvature very close to zero, where $\kappa_{max} \approx -\kappa_{min}$, the principal directions might be well-defined and stable, while the *absolute* principal directions $\mathbf{a}_{max}$ and $\mathbf{a}_{min}$ "interchange", in extreme cases erratically switching between the two principal directions [NPPZ12]. This extreme is illustrated here ($\mathbf{a}_{max}$ on a minimal surface with light noise added). The smooth regions typically do not extend across the curves where these switches happen ($\kappa^g \gg 0$). In areas with such discontinuities our approach may thus underestimate the significance intuitively expected. While we are interested specifically in the use of the smooth regions for alignment to the *absolute* direction $\mathbf{a}_{min}$ (cf. Section 4.2), in other cases it may thus be advisable to define κ^g based on the non-absolute directions, so the smooth regions do not get partitioned by the switching.

While our method itself only has a single intuitive angle parameter, further parameters inevitably come into play when considering the big picture. For instance, prior to application of our method, principal direction estimators may need to be used. Afterwards, e.g., parametrization objectives need to be tuned to get the desired balance between alignment and isometry, etc. Reducing the effort that needs to be spent in this regard would be of great value, but probably requires focusing on concrete application scenarios rather than the abstract generic setting.
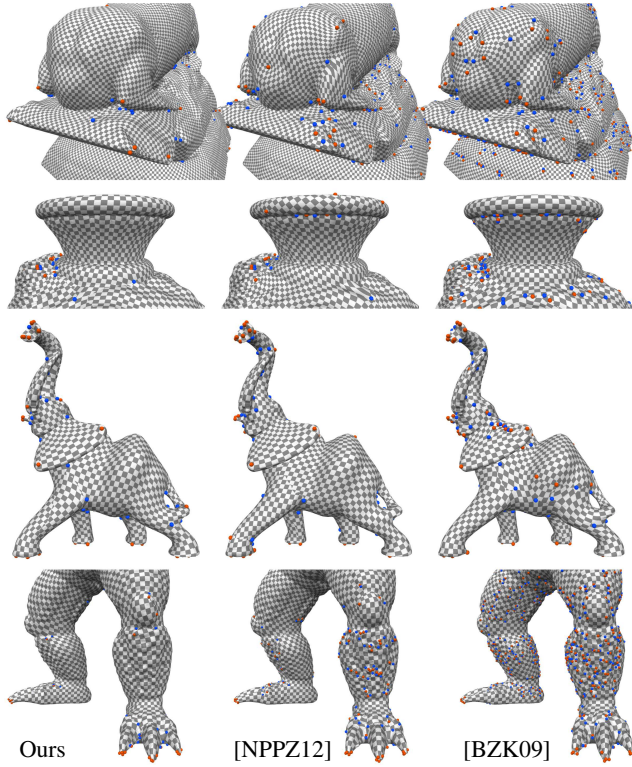
Figure 10: *Global parametrizations aligned using weight fields determined by different methods: ours, [NPPZ12], and [BZK09].*
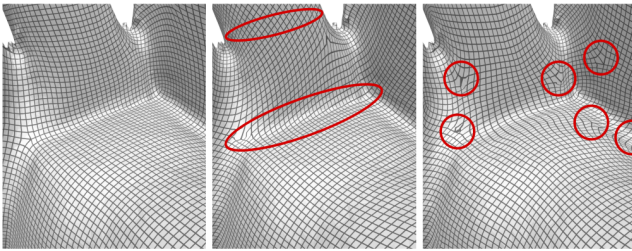

Figure 11: *Parametrization based on [BZK09], with differing alignment weight fields. Left: our method. Middle: [NPPZ12]. Right: [BZK09]. Due to the lack of scale-invariance, in the middle and on the right some obvious features are not aligned to, or some unimportant features are detected, inducing excess singularities.*

Figure 12: *Visualization of isometry deviation (stretch and shearing) of the parametrizations from Figure 10. As can be seen, the significantly higher number of singularities induced by the feature detectors of [BZK09] or [NPPZ12] does not serve isometry improvement. The singularities are truly spurious.*

Acknowledgments

References

[ACSD*03] ALLIEZ P., COHEN-STEINER D., DEVILLERS O., LÉVY B., DESBRUN M.: Anisotropic polygonal remeshing. *ACM Trans. Graph. 22*, 3 (2003), 485–493. 1, 4, 5
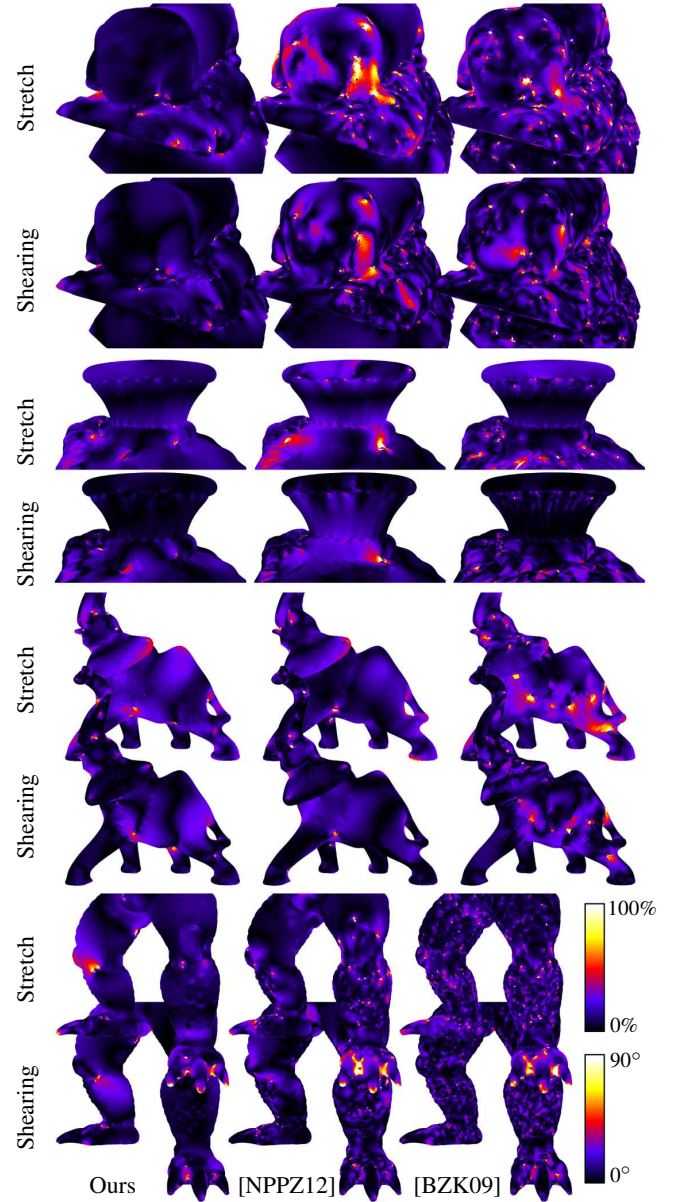
[BCE*13] BOMMES D., CAMPEN M., EBKE H.-C., ALLIEZ P., KOBBELT L.: Integer-grid maps for reliable quad meshing. *ACM Trans. Graph. 32*, 4 (2013), 98:1–98:12. 1, 2, 8

[BK01] BOTSCH M., KOBBELT L.: Resampling feature regions in polygonal meshes for surface anti-aliasing. *Comput. Graph. Forum 20*, 3 (2001), 402–410. 1

[BLK11] BOMMES D., LEMPFER T., KOBBELT L.: Global structure optimization of quadrilateral meshes. *Comput. Graph. Forum 30*, 2 (2011), 375–384. 7

[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). 1

[BZK09] BOMMES D., ZIMMER H., KOBBELT L.: Mixed-integer quadrangulation. *ACM Trans. Graph. 28*, 3 (2009). 1, 2, 3, 4, 6, 7, 8, 9

[CBK15] CAMPEN M., BOMMES D., KOBBELT L.: Quantized global parametrization. *ACM Trans. Graph. 34*, 6 (2015). 1

[CDS10] CRANE K., DESBRUN M., SCHRÖDER P.: Trivial connections on discrete surfaces. *Comput. Graph. Forum 29*, 5 (2010), 1525–1533. 6

[CK14a] CAMPEN M., KOBBELT L.: Dual strip weaving: Interactive design of quad layouts using elastica strips. *ACM Trans. Graph. 33*, 6 (2014). 4, 7

[CK14b] CAMPEN M., KOBBELT L.: Quad layout embedding via aligned parameterization. *Comput. Graph. Forum 33*, 8 (2014). 1, 8

[CSM03] COHEN-STEINER D., MORVAN J.-M.: Restricted delaunay triangulations and normal cycle. In *Proc. Symp. Comp. Geom.* (2003), SCG '03, pp. 312–321. 4

[D'A00] D'AZEVEDO E. F.: Are bilinear quadrilaterals better than linear triangles? *J. Sci. Comput. 22*, 1 (2000), 198–217. 1

[dC76] DO CARMO M. P.: *Differential Geometry of Curves and Surfaces*. Prentice-Hall, Englewood Cliffs, NJ, 1976. 4

[DK12] DAROM T., KELLER Y.: Scale-invariant features for 3-d mesh models. *IEEE Trans. Img. Proc. 21*, 5 (2012), 2758–2769. 3

[DVPSH14] DIAMANTI O., VAXMAN A., PANOZZO D., SORKINE-HORNUNG O.: Designing *N*-PolyVector fields with complex polynomials. *Comput. Graph. Forum 33*, 5 (2014). 6

[DVPSH15] DIAMANTI O., VAXMAN A., PANOZZO D., SORKINE-HORNUNG O.: Integrable polyvector fields. *ACM Trans. Graph. 34*, 4 (2015). 8

[ECBK14] EBKE H.-C., CAMPEN M., BOMMES D., KOBBELT L.: Level-of-detail quad meshing. *ACM Trans. Graph. 33*, 6 (2014). 1, 8

[GG04] GELFAND N., GUIBAS L. J.: Shape segmentation using local slippage analysis. In *Proc. SGP '04* (2004), pp. 214–223. 4

[HPW05] HILDEBRANDT K., POLTHIER K., WARDETZKY M.: Smooth feature lines on surface meshes. In *Proc. SGP '05* (2005), pp. 85–90. 3, 4

[JTPSH15] JAKOB W., TARINI M., PANOZZO D., SORKINE-HORNUNG O.: Instant field-aligned meshes. *ACM Trans. Graph. 34*, 6 (2015), 189:1–189:15. 1, 4

[KCPS13] KNÖPPEL F., CRANE K., PINKALL U., SCHRÖDER P.: Globally optimal direction fields. *ACM Trans. Graph. 32*, 4 (2013). 4, 8

[KMZ11] KOVACS D., MYLES A., ZORIN D.: Anisotropic quadrangulation. *Comp. Aided Geom. Design 28*, 8 (2011), 449–462. 1

[KNP07] KÄLBERER F., NIESER M., POLTHIER K.: Quadcover - surface parameterization using branched coverings. *Comput. Graph. Forum 26*, 3 (2007), 375–384. 1, 3, 4, 7

[KVSM11] KUDELSKI D., VISEUR S., SCROFANI G., MARI J.: Feature line extraction on meshes through vertex marking and 2d topological operators. *Int. J. Image Graphics 11*, 4 (2011), 531–548. 4

[LRL06] LI W.-C., RAY N., LÉVY B.: Automatic and interactive mesh to t-spline conversion. In *Proc. SGP '06* (2006), pp. 191–200. 1

[LXW*11] LIU Y., XU W., WANG J., ZHU L., GUO B., CHEN F., WANG G.: General planar quadrilateral mesh design using conjugate direction field. *ACM Trans. Graph. 30*, 6 (2011), 140:1–140:10. 1

[MK04] MARINOV M., KOBBELT L.: Direct anisotropic quad-dominant remeshing. In *Proc. Pacific Graphics '04* (2004), pp. 207–216. 1

[MOG09] MÉRIGOT Q., OVSJANIKOV M., GUIBAS L.: Robust voronoi-based curvature and feature estimation. In *SIAM/ACM Joint Conf. on Geom. and Phys. Modeling* (2009), SPM '09, pp. 1–12. 4

[MPKZ10] MYLES A., PIETRONI N., KOVACS D., ZORIN D.: Feature-aligned T-meshes. *ACM Trans. Graph. 29*, 4 (2010). 1, 2, 7

[MPS*04] MORTARA M., PATANÉ G., SPAGNUOLO M., FALCIDIENO B., ROSSIGNAC J.: Blowing bubbles for multi-scale analysis and decomposition of triangle meshes. *Algorithmica 38*, 1 (2004), 227–248. 3

[MPZ14] MYLES A., PIETRONI N., ZORIN D.: Robust field-aligned global parametrization. *ACM Trans. Graph. 33*, 4 (2014). 1

[MZ13] MYLES A., ZORIN D.: Controlled-distortion constrained global parametrization. *ACM Trans. Graph. 32*, 4 (2013), 105:1–105:14. 6, 8

[NPPZ12] NIESER M., PALACIOS J., POLTHIER K., ZHANG E.: Hexagonal global parameterization of arbitrary surfaces. *IEEE Trans. Vis. Comput. Graph. 18*, 6 (2012), 865–878. 3, 4, 8, 9

[NSP10] NIESER M., SCHULZ C., POLTHIER K.: Patch layout from feature graphs. *Computer-Aided Design 42*, 3 (2010), 213–220. 3, 4

[OBS04] OHTAKE Y., BELYAEV A. G., SEIDEL H.: Ridge-valley lines on meshes via implicit surface fitting. *ACM Trans. Graph. 23*, 3 (2004), 609–612. 3

[PKG03] PAULY M., KEISER R., GROSS M.: Multi-scale feature extraction on point-sampled surfaces. *Comput. Graph. Forum 22*, 3 (2003), 281–289. 2

[PPTSH14] PANOZZO D., PUPPO E., TARINI M., SORKINE-HORNUNG O.: Frame fields: Anisotropic and non-orthogonal cross fields. *ACM Trans. Graph. 33*, 4 (2014). 1, 8

[PTSZ11] PIETRONI N., TARINI M., SORKINE O., ZORIN D.: Global parametrization of range image sets. *ACM Trans. Graph. 30*, 6 (2011), 149:1–149:10. 1

[PZ07] PALACIOS J., ZHANG E.: Rotational symmetry field design on surfaces. *ACM Trans. Graph. 26*, 3 (2007). 6

[RKS00] RÖSSL C., KOBBELT L., SEIDEL H.-P.: Extraction of feature lines on triangulated surfaces using morphological operators. In *Proc. AAAI Symp. Smart Graph.* (2000), pp. 71–75. 4

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

[RVAL09] RAY N., VALLET B., ALONSO L., LEVY B.: Geometry-aware direction field processing. *ACM Trans. Graph. 29*, 1 (2009), 1:1–1:11. 6

[RVLL08] RAY N., VALLET B., LI W. C., LÉVY B.: N-symmetry direction field design. *ACM Trans. Graph. 27* (2008), 10:1–10:13. 6

[Spi79] SPIVAK M.: *A comprehensive introduction to differential geometry, Vol. II (2nd edition)*. Publish or Perish, inc., 1979. 4

[SWK07] SCHNABEL R., WAHL R., KLEIN R.: Efficient RANSAC for point-cloud shape detection. *Comput. Graph. Forum 26*, 2 (2007), 214–226. 4

[SWPL08] SAHNER J., WEBER B., PROHASKA S., LAMECKER H.: Extraction of feature lines on surface meshes based on discrete morse theory. *Comput. Graph. Forum 27*, 3 (2008), 735–742. 4

[Thi96] THIRION J.: The extremal mesh and the understanding of 3d surfaces. *Int. J. Comput. Vision 19*, 2 (1996), 115–128. 3

[VCD*16] VAXMAN A., CAMPEN M., DIAMANTI O., PANOZZO D., BOMMES D., HILDEBRANDT K., BEN-CHEN M.: Directional field synthesis, design, and processing. *Comput. Graph. Forum 35*, 2 (2016). 3

[WG09] WEINKAUF T., GÜNTHER D.: Separatrix persistence: Extraction of salient edges on surfaces using topological methods. *Comput. Graph. Forum 28*, 5 (2009), 1519–1528. 3, 4

[WHL*13] WANG S., HOU T., LI S., SU Z., QIN H.: Anisotropic elliptic pdes for feature classification. *IEEE Trans. Vis. Comput. Graph. 19*, 10 (2013), 1606–1618. 4

[WK05] WU J., KOBBELT L.: Structure recovery via hybrid variational surface approximation. *Comp. Graph. Forum 24*, 3 (2005), 277–284. 4

[YBS05] YOSHIZAWA S., BELYAEV A. G., SEIDEL H.: Fast and robust detection of crest lines on meshes. In *Proc. Symp. Solid and Physical Modeling '05* (2005), pp. 227–232. 3