Level-of-Detail Quad Meshing

Hans-Christian Ebke, Marcel Campen, David Bommes, Leif Kobbelt

HAL Id: hal-01067289

https://inria.hal.science/hal-01067289

Submitted on 23 Sep 2014

Level-of-Detail Quad Meshing

Hans-Christian Ebke[1][*]   Marcel Campen[1][*]   David Bommes[1][2][*]   Leif Kobbelt[1][*]

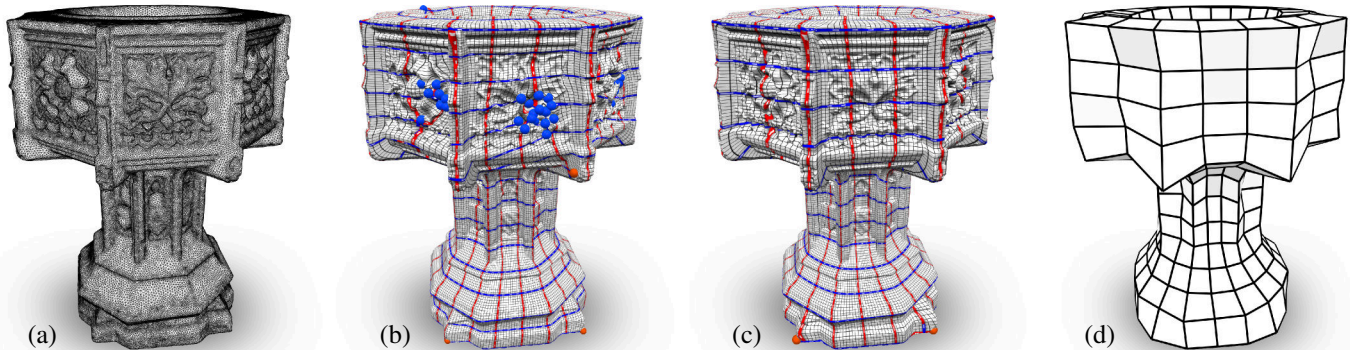[1]RWTH Aachen University   [2]INRIA Sophia Antipolis

Figure 1: *(a) Triangular surface mesh of a font acquired with a laser scanner. The mesh exhibits numerous small scale artifacts and handles and is of (the incorrect) genus 8. (b) Parametrization based quad meshing at a coarse target edge length results in a heavily distorted parametrization with undesired degeneracies and too many singularities too close to one another, even though a state of the art technique for detail suppression [Ray et al. 2009] has been used. (c) Using our method, a parametrization reproducing only features suitable for the target edge length is computed. (d) The resulting quad mesh has the correct genus 0.*

Abstract

The most effective and popular tools for obtaining feature aligned quad meshes from triangular input meshes are based on cross field guided parametrization. These methods are incarnations of a conceptual three-step pipeline: (1) cross field computation, (2) field-guided surface parametrization, (3) quad mesh extraction. While in most meshing scenarios the user prescribes a desired target quad size or edge length, this information is typically taken into account from step 2 onwards only, but not in the cross field computation step. This turns into a problem in the presence of small scale geometric or topological features or noise in the input mesh: closely placed singularities are induced in the cross field, which are not properly reproducible by vertices in a quad mesh with the prescribed edge length, causing severe distortions or even failure of the meshing algorithm. We reformulate the construction of cross fields as well as field-guided parametrizations in a scale-aware manner which effectively suppresses densely spaced features and noise of geometric as well as topological kind. Dominant large-scale features are adequately preserved in the output by relying on the unaltered input mesh as the computational domain.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling

*[*]e-mail:{ebke,campen,bommes,kobbelt}@cs.rwth-aachen.de

Keywords: guiding fields, quad meshing

Links: ◆DL 📄PDF

1 Introduction

Automatic quad remeshing remains a topic of great interest and importance. This is especially true as the proliferation of geometry acquisition equipment progresses. For the resulting wealth of raw data, powerful geometric processing methods are required to refine it into quad meshes suitable for further use and manipulation. Especially field-guided parametrization-based quad meshing methods such as Mixed-Integer Quadrangulation [Bommes et al. 2009], QuadCover [Kälberer et al. 2007], and Periodic Global Parametrization [Ray et al. 2006] have proven to be powerful and versatile. This family of methods typically follows a three-stage approach as illustrated in Figure 2 (a): First a cross field is constructed on the input surface which defines guiding information for the quad element orientation as well as singularity placement. Next, the surface is parametrized into an integer grid map (as defined in [Bommes et al. 2013a]) so that the canonical integer grid in the parameter domain induces a quad mesh on the input geometry. Here, the singularities in the guiding field translate into irregular vertices. An actual quad mesh is extracted from the parametrization in the final step [Ebke et al. 2013].

Through the guiding field, these methods allow for a great amount of user control and since both, the computation of the guiding field and of the parametrization, can be formulated as well behaved optimization problems [Ray et al. 2008; Bommes et al. 2009], good quality solutions can be obtained efficiently. Thanks to these properties, these methods not only enjoy popularity amongst researchers but are also recently finding adoption in modeling and CAD software products.
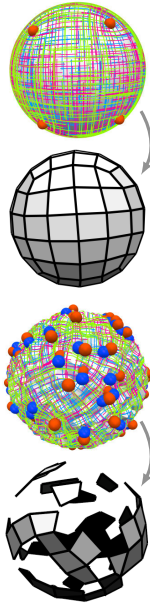
The state of the art methods all assume well behaved input geometry, in the sense that it is noise-free, piecewise smooth, as well as not too detailed nor too feature-rich. Small scale detail, whether intentional ("features") or artificial ("noise"), whether of geomet-

ric or topological (small handles) kind, causes severe problems. In the best case, such suboptimal input results in distortions (i.e. badly shaped quads, badly aligned edges, too many and/or densely spaced irregular vertices), in the worst case the algorithms fail entirely or produce incomplete output meshes. This is illustrated here on a sphere model, where the input mesh on the top is well-behaved, the one on the bottom contains noise.

In practice this turns out to be a significant problem as input geometry is commonly not perfectly well-behaved. This is especially true for real world geometry acquired through laser scanning and related techniques.

There are two different, competing paradigms that can be followed in order to address this problem:

1. pre-process the input geometry in order to make it conform to the requirements of the quad meshing algorithms, or

2. make the meshing algorithms capable of adequately dealing with ill-behaved geometry.

One approach following Paradigm 1 is manual pre-processing, i.e. editing of the input mesh in order to make it well-behaved. For the lack of viable alternatives in many scenarios, this (often tedious) workflow is not uncommon. Where automation is desired or required one could simply treat all types of geometric detail as noise and apply a smoothing algorithm to the input geometry, with the goal of removing or attenuating it. Such approaches can lead to acceptable results—if a suitable choice from the plethora of smoothing methods (ranging from simple Laplacian smoothing [Taubin 1995; Desbrun et al. 1999] to such advanced approaches as the recently introduced conformal Wilmore flow based methods [Crane et al. 2013]) is made. Unfortunately, there is no generally accepted "best" method. Which method to pick depends on which properties are most important for a specific scenario: volume preservation, feature preservation/restoration, few additional parameters, suppression of only high frequencies or of all frequencies alike, etc.

Additionally, since unwanted handles ("topological noise") remain persistent under geometric smoothing, all of the mentioned methods may have to be accompanied by topological mesh repair algorithms (e.g. [Guskov and Wood 2001; El-Sana and Varshney 1997; Zhou et al. 2007; Bischoff et al. 2005]) which make such an approach even more challenging to implement and to tune, impact the run time and potentially introduce new artifacts as outlined in [Attene et al. 2013]. This variation of the generic quad meshing pipeline is illustrated in Figure 2 (b).

In the light of these complications implied by Paradigm 1 we follow Paradigm 2 and introduce a general solution to make parametrization based quad meshing algorithms perform well on ill-behaved input geometry. Our approach requires no additional parameters, which would need model or application dependent tuning. Further, it is non-destructive in the sense that it uses the original, unaltered input mesh as its computational domain which allows for high geometric fidelity of the output mesh. Finally, our approach is simpler to implement and more efficient than the advanced additional methods required by Paradigm 1. Figure 2 (d) illustrates the conceptual pipeline behind our approach.

1.1 Related Work

Our work builds on previous research in the areas of quad remeshing and the construction and processing of cross fields on surfaces.
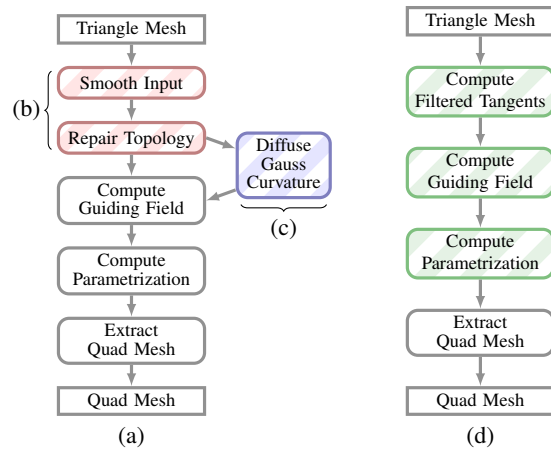

Figure 2: *Omitting the colored boxes, the left side (a) illustrates the generic parametrization based quad meshing pipeline. The red additions (b) reflect Paradigm 1 to deal with ill-behaved input meshes. The blue addition (c) reflects the method from [Ray et al. 2009] which may or may not be combined with (b). The right side (d) illustrates our approach which is in line with Paradigm 2.*

1.1.1 Quad Remeshing

An overview of different quad remeshing methodologies can be found in the survey [Bommes et al. 2013b]. Our solution applies to the class of field-guided parametrization-based methods, first proposed in [Ray et al. 2006], subsequently improved in [Kälberer et al. 2007] and [Bommes et al. 2009], and extended in [Kovacs et al. 2011; Panozzo et al. 2014]. In such methods, the overall quad remeshing problem is split into three sub-steps, namely cross-field computation, integer-grid parametrization and quad mesh extraction out of the parametrization. For the last step, recently a robust method has been proposed in [Ebke et al. 2013] whose specific properties we exploit in Section 6. Furthermore, techniques to increase robustness in this context were introduced [Bommes et al. 2013a; Lipman 2012; Myles et al. 2014].

Since the field-guided pipeline proved to be powerful, generalizations to different input data like point clouds [Li et al. 2011] or range scans [Pietroni et al. 2011] were developed as well. Our approach can similarly be understood as a generalization of the above methods to the case of input geometry that contains detail smaller than the desired resolution of the output quad mesh.

1.1.2 Cross Fields

Apart from complete quad remeshing pipelines there are several works that solely focus on the task of cross field generation [Hertzmann and Zorin 2000]. Some rely on prescribed singularities [Palacios and Zhang 2007; Ray et al. 2008; Crane et al. 2010; Lai et al. 2010], others determine a suitable singularity configuration automatically [Bommes et al. 2009; Panozzo et al. 2012; Knöppel et al. 2013; Diamanti et al. 2014]. All these methods search for the smoothest cross field (in terms of some discrete field curvature notion) potentially subject to certain constraints. In addition, there are methods which only compute a singularity configuration but no cross field [Ben-Chen et al. 2008; Springborn et al. 2008; Myles and Zorin 2012].

Since none of these methods incorporate an explicit concept of scale, noise and small scale detail in the geometry, they necessitate numerous singularities within the smoothest available cross field.
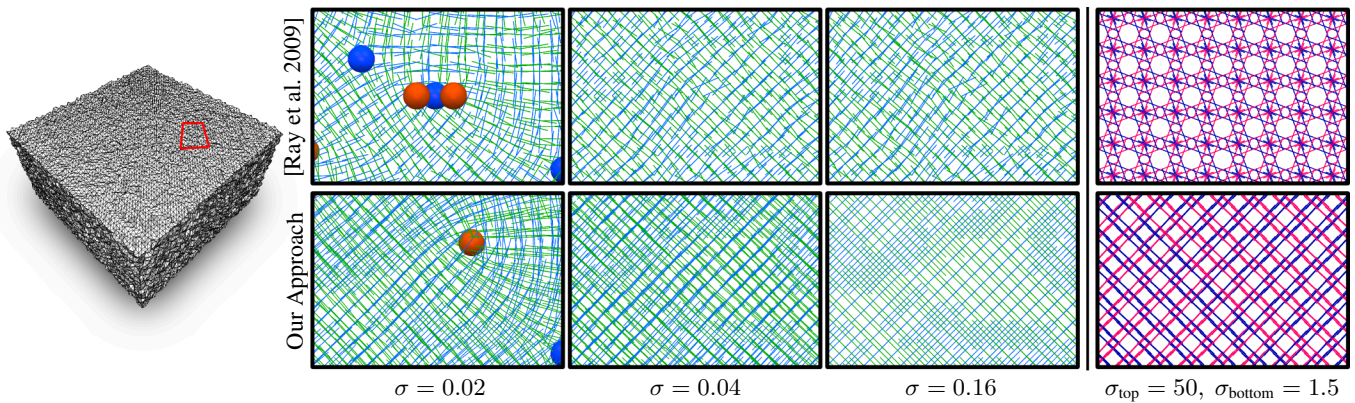
Figure 3: *The two rows of pictures show an orthographic view of different cross fields for the same section (red frame) of the cuboid with geometric noise depicted on the left. As we increase the standard deviation σ of the Gaussian kernel used for the Gaussian curvature smoothing (top row, [Ray et al. 2009]) or for the normal smoothing (bottom row, our approach) the cross fields become smoother. While using our approach (bottom) a perfectly smooth, straight cross field is obtained for increasing σ, it remains imperfect for any σ when using the approach of [Ray et al. 2009] (top) due to the specific cross orthonormality concept. Our crosses, however, are defined in filtered tangent planes (which here converge to the viewing plane). The pictures at the far right demonstrate the behavior for another mesh with periodic detail structures rather than noise. Again, a straight cross field is constructed by our method (bottom), while the cross field computed using [Ray et al. 2009] (top) exhibits severe repetitive distortions.*

The only manner in which some of these methods take some form of scale into account is through the adjustable radius of the employed principal curvature direction estimators that provide initial directions. The further processing, modification, smoothing, interpolation, etc., is then, however, again performed directly at the scale of the input mesh resolution.

The cross field related technique presented in [Ray et al. 2009] can be seen as a first approach into the direction of Paradigm 2 introduced above. In that work, the cross field structure is controlled by first smoothing the scalar field representing the Gaussian curvature of the input mesh, and then constructing a matching tangential transport with respect to which the smoothness of a cross field can be measured. The amount of Gaussian curvature smoothing influences the structure of the resulting cross field and, as demonstrated by the authors, this allows to effectively reduce the number of singularities which are usually caused by small scale details in the input surface. However, this method only addresses the cross field construction step, where tangential transport suffices to measure smoothness. For the quad meshing scenario also the parametrization step needs to be considered. Furthermore, this method uses crosses which are orthonormal with respect to the original surface while the field smoothness is measured with respect to the tangential transport of the intrinsically smoothed surface. This discrepancy can lead to distortions (cf. Figure 3) which carry over to the field-guided parametrization. Finally, note that only geometric ill-behavedness is addressed: the Gaussian curvature induced by small handles in the input geometry cannot be cancelled out locally through smoothing. As a result, the corresponding singularities do not vanish but are only slightly dispersed. This problem is illustrated in Figure 4.

1.2 Contribution

To overcome the problems of the existing methods we propose a solution which equips the input mesh with an alternative (non-orthogonal) normal field that respects the desired target mesh resolution and suppresses detail too small to be adequately reproduced. Varying target resolution (a so-called *sizing field*) can be taken into account as well. All relevant aspects (like field smoothness, field orthogonality, field normality, parametrization fairness, and con-

straints) are then expressed and evaluated with respect to this setup based on alternative normals and tangent planes. This way we are able to effectively prevent small scale geometric detail from being carried over into the guiding field or the parametrization, and thus into the resulting quad mesh.

When constructing the alternative normal field we go so far to even make use of local inversions, where areas in the parameter domain become negative. This happens in such a manner that ultimately small scale folds, protrusions, and even topological handles are "ironed over" and flattened (thus appropriately cancelled) rather than unfolded into the parametric domain. This prevents distortions and problematic singularities that would otherwise occur. Figure 5 illustrates this idea.

Due to this novel robustness to ill-behaved input, the need for expensive pre-processing is eliminated.

2 Concept

The idea behind our approach may be pictured using the following conceptual image:

State-of-the-art approaches construct cross fields and parametrizations which are smooth with respect to the standard metric on the (ill-behaved) input surface. Our approach can be thought of as
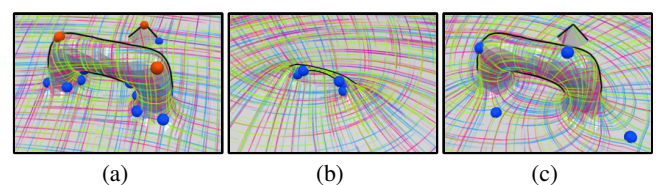

Figure 4: *(a) A small scale handle and the cross field singularities surrounding it (b) remain persistent under smoothing. (c) Using the method from [Ray et al. 2009] the singularities are dispersed slightly but persist. The small spike in the background is effectively suppressed using either method. As shown in Figure 12 this configuration is properly handled by our method.*
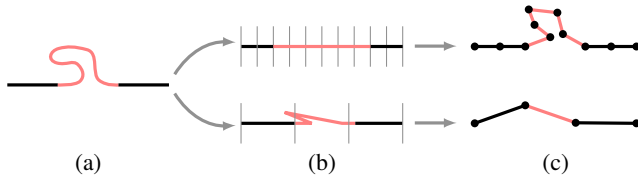
Figure 5: *Geometry with small scale features (a) is parametrized (b) in order to extract a mesh from the parametrization (c). The upper row illustrates the process with a small target edge length where the feature (red) is reproduced in the output mesh (c top). In the bottom row a large target edge length is chosen. Our method deliberately creates a partially flipped parametrization (b bottom). Flipped and regular elements in the parametrization cancel out one another so that the small scale feature is not reproduced and does not introduce length distortion in the output (c bottom).*

Figure 6: *Illustration of the well-behaved shell (grey) with its smooth normal field (red). Measured with respect to this field, segment a is much shorter than segment b, although they are of the same length in the metric of the original surface (blue). Segment c even has a negative length, basically cancelling out the fold. The same holds for segments d and e, effectively cancelling out the small handle.*

instead measuring smoothness (and orthogonality, normality, etc.) within a well-behaved shell surrounding the surface. This is illustrated using a 2D example in Figure 6. In this shell, lengths (and in 3D also angles) are not measured in a standard, Euclidean manner but, conceptually, within a subspace orthogonal to a vector field (red) – which we construct as a discrete filtered version of the input's normal field (cf. Section 3). To achieve this, we basically express all measurements with respect to local projections of the original geometry into the filtered tangent planes defined by the filtered normals. This projection can even affect the orientation. In Figure 6 the intriguing properties of this concept are illustrated. Most importantly, we also define crosses to live (and be orthonormal) on the filtered tangent planes, rather than on the surface itself.

All formulations are such that they are equivalent to traditional formulations for the case that the filtered normal field is orthogonal to the given surface.

It is worth noting that the projections of the mesh's individual faces into their respective filtered tangent planes do not necessarily constitute a globally consistent smooth surface. In our approach we can, however, conveniently restrict ourselves to local considerations only – of individual triangles (for the parametrization), pairs of triangles (for cross field smoothness), or 1-rings (for curvature evaluation). Therefore, neither the conceptual shell nor a complete hypothetical surface orthogonal to the filtered normal field actually need to be constructed in our method.

Note that in certain cases (when there are no folds and no handles), smoothing of the input mesh as discussed in Section 1 can be considered as constructing a surface which is (more or less) close to orthogonal to the filtered normal field – but whose normal field is typically less smooth as it inherently obeys the integrability condition. From a theoretical perspective this global consistency of the surface is appealing. However, it is unclear whether this would lead to any practical advantage. We only identified and observed disadvantages (in addition to the restricted applicability) as discussed in Sections 1 and 7.

2.1 Setup

Our approach is quite generic and could be used in conjunction with various field generation methods and various parametrization strategies. We demonstrate it here using the period-jump based cross field representation proposed in [Li et al. 2006] and the field-guided parametrization functional proposed in [Ray et al. 2006; Bommes et al. 2009], as these are quite prototypical: they have been used in numerous follow-up works like [Ray et al. 2008; Kälberer et al. 2007; Bommes et al. 2009; Crane et al. 2010; Panozzo et al.
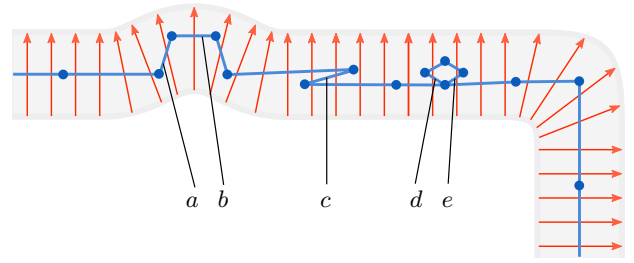
2012; Bommes et al. 2013a; Ray et al. 2009; Campen and Kobbelt 2014b; Li et al. 2011; Pietroni et al. 2011]. In particular, we make use of the unified mixed-integer formulation introduced in [Bommes et al. 2009][1].

We thus use the following generic quad remeshing pipeline for demonstration:

1. estimation (and filtering) of principal curvature directions, or manual specification of constraint directions,

2. construction of a smooth cross field based on these directions,

3. parametrization of the input mesh guided by the cross field,

4. extraction of the quad mesh defined by the canonical integer grid in the parameter domain.

All four steps are affected by our approach, most importantly of course step 2 and 3. In the following we detail the construction of the filtered normal field (Section 3) and then explain how to reformulate the individual steps of the pipeline accordingly.

3 Smooth Normals

For a meshed surface $\mathcal{M} = (V, E, F)$ with vertices V, edges E, faces F, we compute an alternative, filtered normal field $\hat{N} = \{\hat{\mathbf{n}}_1, \ldots, \hat{\mathbf{n}}_{|F|}\}$ (which is detached from the actual surface, i.e. not necessarily orthogonal). The key idea behind this is that geometric features are reflected by high variance in the original normal field N. In order to prevent sets of features which are too densely spaced to be representable by the desired target element size from carrying over into the cross-field, we apply a generalized Weierstrass transform to this normal field. I.e. we convolve it with a suitably sized Gaussian kernel G_σ, to obtain a new normal field $\hat{N}$ where densely spaced features are merged (cf. Figure 7):

$$\hat{N}_\sigma(\mathbf{p}) = (N * G_\sigma)(\mathbf{p}) = \int_{\mathcal{M}} N(\mathbf{q}) G_\sigma(\mathrm{dist}(\mathbf{p} - \mathbf{q})) \mathrm{d}\mathbf{q}$$

with

$$G_\sigma(x) = \frac{1}{2\pi\sigma^2} e^{-x^2/(2\sigma^2)}$$

---

[1]A variation of the globally optimal cross field construction approach of [Knöppel et al. 2013] could also be used, but does not immediately allow for, e.g. direct control over singularity positions, indices, or sparse alignment constraints – which is often of interest in the quad remeshing context.
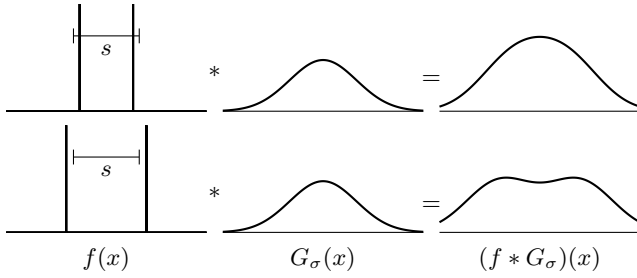
Figure 7: *Top: features in the normal field (modeled as Dirac impulses) closer to one another than the target edge length s get merged into one maximum when convolved with a Gaussian kernel with $\sigma \geq s/2$. Bottom: features with wider spacing are reproduced as two distinct maxima in the convolved field.*

where $\text{dist}(\mathbf{p} - \mathbf{q})$ signifies the geodesic distance between $\mathbf{p}$ and $\mathbf{q}$ on $\mathcal{M}$. Note that $\hat{N}_\sigma$ is not a unit vector field in general. Thus, we renormalize it prior to any further use. In practice, we truncate the Gaussian kernel by integrating over $B_r(\mathbf{p}) \subset \mathcal{M}$, which is the neighborhood of $\mathbf{p}$ up to some geodesic distance r. A choice of $r = 2\sigma$ proved sufficient.

For the purpose of discretization we assume a constant normal per face f, evaluated at its barycenter $\mathbf{c}(f)$, and get

$$\hat{N}_\sigma(f) = \sum_{f' \in B_{2\sigma}(f)} N(f') \int_{f'} G_\sigma(\text{dist}(\mathbf{c}(f) - \mathbf{q})) d\mathbf{q}.$$

We further approximate the integral using a sample in the barycenter $\mathbf{c}(f')$ of the faces f' and obtain

$$\hat{N}_\sigma(f) \approx \sum_{f' \in B_{2\sigma}(f)} N(f') G_\sigma(\text{dist}(\mathbf{c}(f) - \mathbf{c}(f'))) A_{f'}$$

where $A_{f'} = A(f' \cap B_{2\sigma}(f))$ is the area of the portion of f' which lies inside the truncated kernel.

Note that, instead of explicitly evaluating the above sum per face, one can also employ diffusion flow to implicitly convolve the normals [Desbrun et al. 1999].
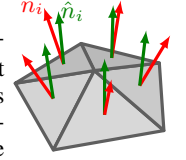
The standard deviation σ controls the breadth of the filter kernel G_σ. We choose σ dependent on the prescribed target edge length s so that features in the normal field closer to one another than s get merged when convolved with G_σ. In order to derive a suitable σ we imagine two infinitely sharp features with distance s on an otherwise flat curve. We model these features as a function $f : \mathbb{R} \to \mathbb{R}$ with two Dirac impulses at $\pm s/2$. As illustrated in Figure 7, if we convolve f with G_σ choosing σ too small, two distinct local maxima persist, and we have a local minimum, i.e. positive curvature, at $x = 0$. Choosing σ sufficiently large we yield negative curvature at $x = 0$ and thus a single maximum:

$$\left. \frac{d^2}{d^2 x^2} \int_{-\infty}^{\infty} f(q) G_\sigma(x - q) dq \right|_{x=0} \leq 0$$

$$\Leftrightarrow \left. \frac{d^2}{d^2 x^2} G_\sigma\left(x - \frac{s}{2}\right) + G_\sigma\left(x + \frac{s}{2}\right) \right|_{x=0} \leq 0$$

$$\Leftrightarrow \sigma(s) \geq \frac{s}{2}.$$

While the choice of σ may be left to the user as a degree of freedom in a setting where more interactivity is desired, we advertise the choice of $\sigma = s/2$ in all settings where an additional parameter

in the quad meshing pipeline is unwanted. In agreement with the theoretical justification, our experiments confirm that this choice of σ generally yields very good results.

We refer to the tangent planes defined by the filtered normals $\hat{n}_i = \hat{N}_\sigma(f_i)$ as *filtered* tangent planes in the following. The filtered tangent planes can be computed at almost no computational overhead as the neighborhood determination has to be done for the curvature estimation (shape operator, etc.) anyway.

It is worth noting that averaging of vertex normals from 1-ring face normals (which can be interpreted as a simplistic variant and special case of our normal filtering) was reported to already show beneficial effects on the cross field structure [Myles et al. 2010].

Sizing Field Instead of targeting a uniform quad element size s, we may also take a varying element sizing field $S : F \to \mathbb{R}^{>0}$ into account if desired. In this case σ varies per face, i.e. we have $\sigma(f)$ (e.g. $\sigma(f) = S(f)/2$) and generalize to
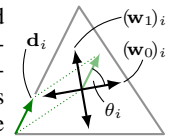
$$\hat{N}_S(f) = \sum_{f' \in B_{2\sigma(f)}(f)} N(f') G_{\sigma(f)}(\text{dist}(\mathbf{c}(f) - \mathbf{c}(f'))) A_{f'}.$$

4 Normal Based Cross Fields

To draw benefit from our filtered tangent planes in the construction of the guiding field for the subsequent parametrization, we derive a formulation for the cross field formalism [Li et al. 2006] based entirely on the filtered normal vectors.

Given a piecewise linear surface $\mathcal{M} = (V, E, F)$ equipped with filtered face normals $\hat{N} = \{\hat{\mathbf{n}}_1, \ldots, \hat{\mathbf{n}}_{|F|}\}$ we define a (discrete) filtered cross field on that surface to consist of four vectors per face, i.e. $\mathbf{w}_k : F \to \mathbb{R}^3$, $k = 0, 1, 2, 3$, such that $\mathbf{w}_k(f_i) \perp \hat{\mathbf{n}}_i$, and for orthonormality require $\|\mathbf{w}_k(f_i)\| = 1$ and $\mathbf{w}_k(f_i) = \left(\text{Rot}_{90}^{\hat{\mathbf{n}}_i}\right)^k \mathbf{w}_0(f_i)$, where $\text{Rot}_{90}^{\hat{\mathbf{n}}}$ is a 90-degree rotation around the axis $\hat{\mathbf{n}}$. We use $(\mathbf{w}_k)_i$ as a shorthand for $\mathbf{w}_k(f_i)$.

We employ the angle based definition introduced in [Li et al. 2006]: given arbitrary reference directions $D = \{\mathbf{d}_1, \ldots, \mathbf{d}_{|F|}\}$, $\mathbf{d}_i \perp \hat{\mathbf{n}}_i$ per triangle, a cross field is uniquely defined by angles $\Theta = \{\theta_1, \ldots, \theta_{|F|}\}$ where θ_i describes the angle between $\mathbf{d}_i$ and $\mathbf{w}_0(f_i)$ in the filtered tangent plane.

This, so far, defines an individual cross per face. Using so-called period jumps $p_{ij} \in \mathbb{Z}$ per edge one specifies which of the four directions in a face corresponds to which of a neighboring face. This flexible association enables the representation of singularities with fractional index [Li et al. 2006].

4.1 Measuring Smoothness

In order to measure the smoothness of the cross field we need a way to measure the deviation γ_{ij} between the two directions θ_i, θ_j of a pair of neighboring faces (i, j). Since each direction is defined through an angle θ with respect to its respective reference direction $\mathbf{d}$, by computing a signed transition angle κ_{ij} between $\mathbf{d}_i$ and $\mathbf{d}_j$ we can compute the deviation as

$$\gamma_{ij} = \theta_i + \kappa_{ij} + \frac{\pi}{2} p_{ij} - \theta_j.$$

κ_{ij} can be computed as the signed angle between $\mathbf{d}_i$ and $\mathbf{d}_j$ after rotating them into a common plane using a hinge map [Li et al. 2006]. While the hinge axis usually is the common edge e_{ij} of the

pair of faces, this does not fit our filtered setup as e_{ij} does not lie in the filtered tangent planes in general. Instead we compute the appropriate hinge axis $\hat{\mathbf{e}}_{ij}$ from the filtered normals:

$$\hat{\mathbf{e}}_{ij} = \hat{\mathbf{n}}_i \times \hat{\mathbf{n}}_j$$

and can then evaluate

$$\kappa_{ij} = \measuredangle(\mathbf{d}_i, \hat{\mathbf{e}}_{ij}) + \measuredangle(\hat{\mathbf{e}}_{ij}, \mathbf{d}_j).$$

Note that when using the original normals, this definition is equivalent to the usual formulation. If $\mathbf{n}_i$ and $\mathbf{n}_j$ are identical, i.e. $\hat{\mathbf{e}}_{ij}$ has zero length, the reference directions already lie in the same plane and $\kappa_{ij} = \measuredangle(\mathbf{d}_i, \mathbf{d}_j)$ is used instead.

With these definitions the discrete field curvature of the filtered cross field can be expressed as [Ray et al. 2008; Bommes et al. 2009]:

$$E_{\text{smooth}} = \sum_{e_{ij} \in E} w_{ij} \gamma_{ij}^2. \tag{1}$$

Note that the weights w_{ij} are simply chosen to be 1 in most related methods, e.g. [Ray et al. 2008; Bommes et al. 2009; Li et al. 2006]. Experiments show that the benefit of using proper weighting (depending on the face geometry) often is minuscule. In fact, for all results presented in this work, uniform weights were used. For completeness we can, however, also adapt the proper dual cotangent weights introduced for this setting in [Crane et al. 2010] to our setup, as we show below.

4.2 Weights

The w_{ij} in Equation (1) account for size and shape differences between faces. [Crane et al. 2010] advocate the use of $w_{ij} = (\cot(\phi_{ij}) + \cot(\phi_{ji}))^{-1}$, where ϕ_{ij} and ϕ_{ji} are the angles of the two sectors opposite to e_{ij}.

In our setting it is appropriate to compute these weights based on the projections of these sectors into the filtered tangent planes. We thus compute the *projected* dual cotangent weights

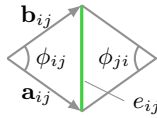$$w_{ij} = (\cot(\phi'_{ij}) + \cot(\phi'_{ji}))^{-1}$$

with $\phi'_{ij} = \measuredangle(\mathbf{a}'_{ij}, \mathbf{b}'_{ij})$ where

$$\mathbf{a}'_{ij} = \mathbf{p}_{\hat{\mathbf{n}}_i}(\mathbf{a}_{ij}) \text{ and } \mathbf{b}'_{ij} = \mathbf{p}_{\hat{\mathbf{n}}_i}(\mathbf{b}_{ij}).$$

Here $\mathbf{a}_{ij}$ and $\mathbf{b}_{ij}$ represent the edge vectors of the sector corresponding to ϕ_{ij} on the *original* input mesh as illustrated on the right. $\mathbf{p}_{\hat{\mathbf{n}}}$ is the projection into the filtered tangent plane along the filtered normal $\hat{\mathbf{n}}$:

$$\mathbf{p}_{\hat{\mathbf{n}}_i}(\mathbf{v}) = (\text{Id} - \hat{\mathbf{n}}_i \hat{\mathbf{n}}_i^\intercal)\mathbf{v}.$$

Thus, ϕ'_{ij} and ϕ'_{ji} are the angles of the corners opposite to e_{ij} in the projected images of f_i and f_j. Note that since each face is projected along its individual filtered normals, in general images of adjacent faces do not match up (e.g. in Figure 1, the average length mismatch of the common edge is 0.18%, the maximum 4.4%). Thus the projected dual cotangent weights do not exacly correspond to an actual surface. Notwithstanding this, they do behave in a plausible way according to all our experiments. Furthermore, they converge to the original dual cotangent weights when the Gaussian kernel size approaches 0 ($\sigma \to 0$), i.e. when the filtered normals are identical to the original normals they are equivalent, and they converge to weights of an actual surface when pairs of adjacent filtered normals converge, e.g. when $\sigma \to \infty$.

4.3 Alignment Constraints

The minimizer of E_{smooth} in Equation 1 gives us a cross field which is smooth but which is not yet aligned along the principal curvature directions (or user defined directions) where desired. To achieve this, we can constrain some of the θ_i to prescribed angles, eliminating these variables from the optimization problem. The directions of (minimal or maximal) principal curvature are typically computed using the Shape Operator [Cohen-Steiner and Morvan 2003] and take the form of a unit vector $\mathbf{m}_i \in \mathbb{R}^3$ for a face f_i. Instead of projecting $\mathbf{m}_i$ onto the face as usual, we project it into the filtered tangent plane to compute the prescribed angle θ_i:

$$\theta_i := \measuredangle(\mathbf{d}_i, \mathbf{p}_{\hat{\mathbf{n}}_i}(\mathbf{m}_i)).$$

4.4 Period Jumps

In order to evaluate E_{smooth} in Equation 1 what is still missing are the period jumps p_{ij}. Period jumps can either be prescribed [Li et al. 2006] (which is equivalent to prescribing singularity positions and indices) or they can be left as a degree of freedom in a mixed integer problem [Bommes et al. 2009] which is the approach we took in our experiments.

4.5 Gaussian Curvature

Once we have computed a minimizer for E_{smooth} we need to evaluate the cross field index $I(v_i)$ for every vertex v_i in order to determine the position and degree of the singularities [Li et al. 2006]:

$$I(v_k) = \frac{1}{2\pi}\left(K(v_k) + \sum_{e_{ij} \in E_k} \kappa_{ij}\right) + \frac{1}{4}\sum_{e_{ij} \in E_k} p_{ij}$$

where E_k specifies the set of edges incident to v_k, and $K(v_k)$ is the Gaussian curvature at vertex v_k. In order to rely exclusively on the filtered normals, we compute the Gaussian curvature from the signed area of the spherical polygon spanned by the filtered normal vectors $\hat{\mathbf{n}}_i$ of the faces incident to v_k [Meek and Walton 2000].

5 Parametrization

The second stage in the three stage pipeline is the computation of a parametrization $\mathbf{g} : \mathcal{M} \to \mathbb{R}^2$ guided by the cross field from the first stage. If $\mathbf{g}$ is an Integer Grid Map [Bommes et al. 2013a] the canonical grid of integer iso-lines in $\mathbb{R}^2$ induces a quad mesh on $\mathcal{M}$ through $\mathbf{g}$.

The parametrization $\mathbf{g}$ is defined on the mesh – the cross field which is supposed to guide the parametriztion, however, lives in the filtered tangent planes. We thus project the crosses from their filtered tangent planes onto the faces along the filtered normals:

$$(\bar{\mathbf{w}}_k)_i = (\mathbf{w}_k)_i - \hat{\mathbf{n}}_i \frac{\mathbf{n}_i^\intercal(\mathbf{w}_k)_i}{\mathbf{n}_i^\intercal \hat{\mathbf{n}}_i}.$$

Here the $\bar{\mathbf{w}}_k$ are the projections of the cross field vectors $\mathbf{w}_k$ which live in the filtered tangent planes.

Note that this results in non-orthonormal crosses whenever $\mathbf{n}_i \neq \hat{\mathbf{n}}_i$. The parametrization functional used in [Bommes et al. 2009] is only suitable for orthonormal crosses. We modify it and compute the parametrization as the minimizer of

$$E_{\text{orient}} = \sum_{f \in F} E_f \mathbf{A}(\hat{f})$$
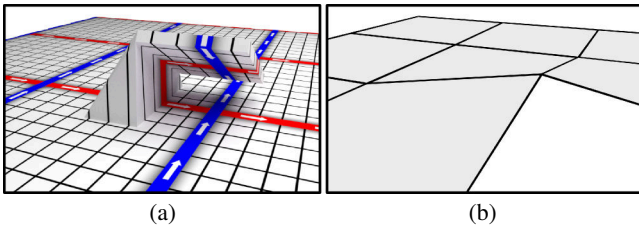
(a)   (b)

Figure 8: *(a) Example of an overlapping small scale detail. As the orientation of the arrows indicates, the parametrization flips locally. (b) After extraction of the quad mesh, flipped and non-flipped areas have cancelled out one another and the geometric influence of the feature is minimal.*

with

$$E_f = \left\| \nabla \mathbf{g} \cdot \left[\bar{\mathbf{w}}_0 \ \bar{\mathbf{w}}_1 \right] - S(f) \operatorname{Id} \right\|^2$$

with respect to the constraints laid out in [Bommes et al. 2009]. Intuitively, the (tangent) cross vectors $\bar{\mathbf{w}}_0$ and $\bar{\mathbf{w}}_1$ are mapped from the faces to the parametrization domain $\mathbb{R}^2$ via the map's differential (the Jacobian $\nabla \mathbf{g}$). There they should ideally coincide with the coordinate axes and have length $S(f)$, as then the integer grid mapped back to the face is aligned with the cross and has the desired spacing $S(f)$. Note that for proper weighting the area of the projected faces $\hat{f}$ is used in E_{orient}.

The variational formulation readily allows for easy modification and extension to include various types of useful hard or soft constraints, e.g. for feature alignment [Bommes et al. 2009] or connectivity [Myles et al. 2010].

6 Quad Mesh Extraction

An important consequence of the non-orthonormal nature of the projected cross-field is that crosses can be flipped. This common case arises when small scale features form overlaps as is illustrated in Figure 5 which, after normal filtering lead to flipped normals. Flipped crosses in E_{orient} promote flipped triangles in the parametrization which lead to so-called fold-over configurations. In principle, fold-overs violate the Integer Grid Map condition and lead to non-quad meshes. If, however, we interpret flipped triangles as having a negative area that cancels out an overlapping positive area, they neutralize the geometric influence of the affected small scale features which is a most welcome effect. Through our choice of the Gaussian kernel breadth $\sigma = s/2$ we only provoke such fold-overs on a scale well below the target edge length s. Such fold-overs are handled in the desired way, cancelling out overlapping regions, by the recently introduced quad extractor QEx [Ebke et al. 2013] which we employ in our approach (cf. Figure 8). This way we are able to reliably and robustly extract quad meshes from the generated parametrizations.

It should be noted that apart from the fold-overs deliberately introduced through flipped crosses, the least-squares parametrization may produce fold-overs in certain situations as detailed in [Bommes et al. 2013a; Ebke et al. 2013]. Such fold-overs can be prevented using a stiffening approach [Bommes et al. 2009] or constraints [Bommes et al. 2013a; Lipman 2012]. Alternatively, they can be admitted and gracefully dealt with using QEx. We went with the latter option in all of our experiments.

7 Results and Comparison

In Section 1.1.2 we made the point that the method in [Ray et al. 2009] merely considers the cross field generation and does not of-
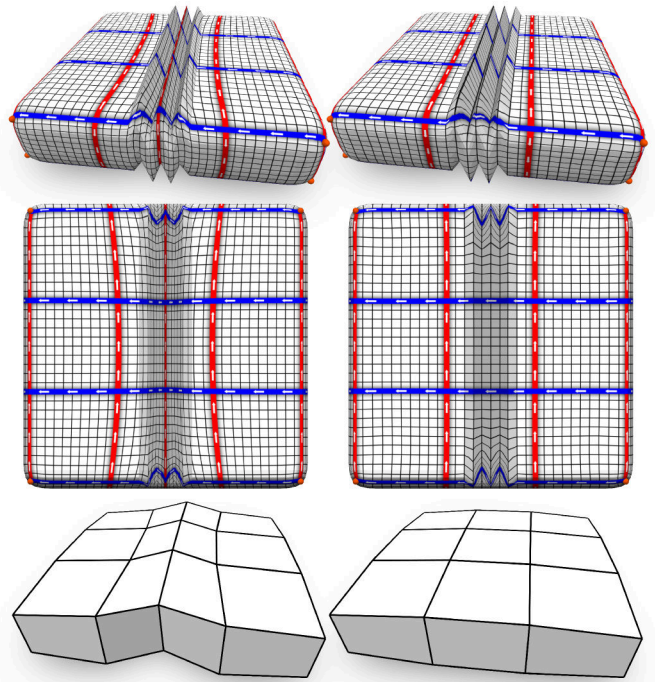

Figure 9: *Two parametrizations and the resulting quad meshes of the same input mesh. In the orthographic view (center row) it becomes apparent that the additional area introduced by the accordion causes a length distortion using the method of [Ray et al. 2009] (left column) wheras using our method the accordion has little effect on the parametrization (right column).*

fer a way to introduce a sense of scale into the parametrization process which can lead to distortions in the output quad mesh. Figure 9 demonstrates this effect. Here, [Ray et al. 2009] successfully manages to suppress singularities at the small scale features but the length distortion caused by the accordion structure is so severe that two quad loops at half the width of the target edge length appear. The parametrization generated by our method remains unaffected by the accordion.

Figure 12 demonstrates how our method is able to extract quad meshes that are unaffected by small scale handles on the input geometry and exemplifies the fact that smoothing approaches as well as the method of [Ray et al. 2009] always reproduce such handles with densely spaced singularities. In Figure 10 we demonstrate the shortcomings of Laplacian surface smoothing in combination with the stock Mixed-Integer Quadrangulation approach on a real world laser scanned mesh with a number of topological and geometrical artifacts.

In Figure 11 we show a number of real-world input meshes quad meshed using our approach. The images of the plain input triangle meshes show the singularities that are generated when computing a non scale-aware crossfield [Ray et al. 2008] which, in all of these examples are so densely spaced that they are only reproducible by excessively fine quad meshes. With our method we were able to extract sensible, watertight quad meshes at a wide range of target edge lengths for all input meshes. Only the BUDDHA mesh represents a special case as explained in Section 8. The BOLTDISK mesh demonstrates the ability of our method to incorporate feature alignment constraints (amongst all other common parametrization constraints).
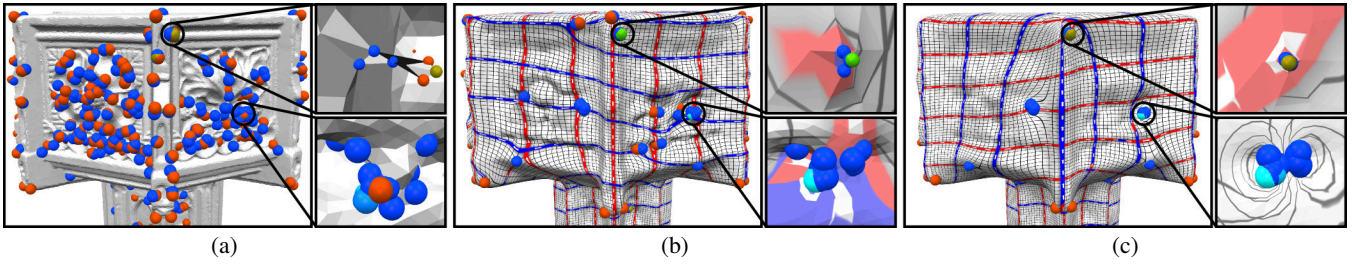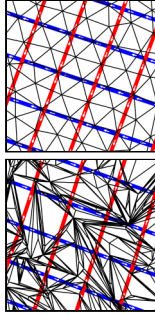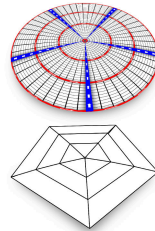
Figure 10: *The FONT mesh (c.f. Figure 1) with (a) 0, (b) 30 and (c) 100 iterations of Laplacian smoothing applied to it. Computing a cross field using the traditional Mixed-Integer Quadrangulation approach yields 1238, 147 and 85 singularities, respectively. The excess of singularities in the unsmoothed version (a) and thus the excessive number of integer variables in the resulting parametrization optimization problem prevents finding a parametrization within reasonable time. The parametrizations of the smoothed versions (b) and (c) suffer from the remaining badly placed singularities as well as from the topological noise. This can be witnessed in the magnifications of a spike (top) and a small handle (bottom). While the spike is manifold, it suffers from severe self-intersections and is thus not completely eliminated even after a considerable amount of smoothing iterations (c). The handles and their accompanying singularities persist after any number of iterations and thus will always ruin the parametrization. Even though the parametrization in (c) roughly looks like a quad mesh on a macroscopic level, a quad mesh cannot be extracted out of it due to the microscopic artifacts around the singularity clusters.*
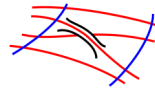
8 Limitations and Future Work

We showed that our method is robust against geometrical and topological artifacts in the input geometry. The input quality in terms of triangle shape is not important either as demonstrated in the images on the right where two planar meshes, one with a mostly isotropic triangulation, the other one with a highly anisotropic triangulation are parametrized using our method and yield almost equivalent results. Also, the STATUETTE input mesh in Figure 11 exhibits highly anisotropic triangles with aspect ratios of up to 502 and inner angles ranging from .2 up to 179.5 degrees. However, the robustness against *holes* in the input mesh is limited: while the normal smoothing, cross field computation and parametrization steps do not have particular problems with holes, they may interrupt an integer iso-line in the parametrization, potentially leading to missing elements in the output. This event becomes more likely as the size of a hole increases relative to the target edge length. I.e. our method does not go so far as to perform implicit hole filling.

While our approach suppresses small handles, small tunnels (which are less likely to appear as an unintended artifact) persist. This is owed to the fact that under normal smoothing the normals of a small handle locally converge towards a flat configuration whereas the normals of a small tunnel do not. This is demonstrated on the right where a small tunnel connecting the two sides of a disk gets reproduced (i.e. the input and the output mesh both have genus 1) even though the target edge length exceeds the diameter of the tunnel by several orders of magnitude.

An assumption of our method is that wanted handles and unwanted handles in the input mesh are well separated in terms of scale. This is because handles with a circumference close to the target edge length may lead to iso-lines which are parallel in the parameter domain tunneling one another on the surface as illustrated on the right. In such a case the extracted mesh may locally be non-manifold. The BUDDHA depicted in Figure 11 is a particularly interesting mesh in this regard since its 100 handles come in almost every size from "microscopic" to "huge". As a result, some tuning of the target edge length was necessary to yield a manifold result.

Our approach disregards small scale features and artifacts by attenuating high-frequencies in the normal field. Note that there is, however, not a direct correspondence between high-frequencies and small scale details: also sharp creases and corners carry high frequencies in the normal field – which are attenuated even if these features are isolated enough to be properly reproduced. While distinct maxima remain at such features (cf. Figure 7) – inducing the desired singularities – they are less pronounced after the filtering. But as we only perform the filtering in the normal field and preserve the input geometry, these features can still be reproduced geometrically correct. Also singularities often still arise on sharp corners instead of somewhere in their vicinity, but there is no general guarantee. For further improvement in this regard it would be desirable to be able to better distinguish between features that are just too small to be reproducible and those that can be captured by the final quad mesh. The use of another, e.g. bilateral or Perona-Malik type of filter could be a first step in this direction. A true distinction is, however, hardly possible already in the guiding field construction stage – it depends too much on the global structure and alignment of the parametrization – so that it seems unlikely that significant improvements in this regard are possible within the common three-stage strategy for field guided parametrization quad remeshing.

Finally, it is worth noting that our approach is not restricted to parametrization based quad meshing methods alone. It promises improved results in any method relying on guiding fields such as the quad domain construction method in [Tarini et al. 2011], the quad layout approach in [Campen et al. 2012; Campen and Kobbelt 2014a] or various non-photo realistic rendering techniques [Hertzmann and Zorin 2000; Umenhoffer et al. 2011]. Also, the extension to anisotropic and non-orthogonal cross fields [Panozzo et al. 2014] should be possible.

9 Conclusion

We presented a method to generate cross fields and in a subsequent step integer grid parametrizations which are unaffected by small scale features and artifacts such as noise, or small handles in the input mesh. The key idea was to express and measure all involved quantities based on a filtered normal field. This resulted in robustness to ill-behaved input, thus eliminating the need for expensive pre-processing. We showed how quad meshing in particular can benefit from our method and there are further applications relying on guiding fields which may very well draw profit from it, too.
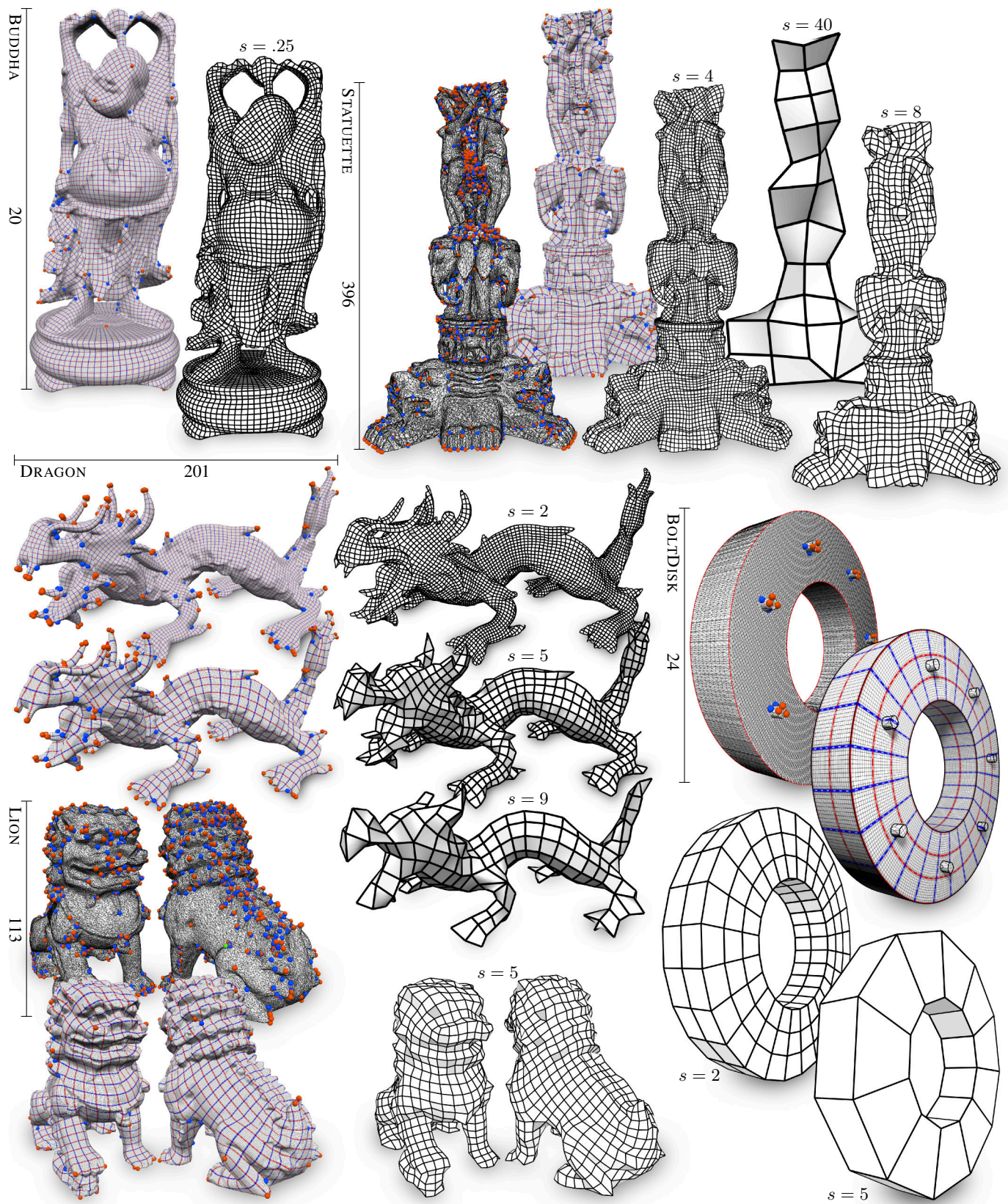
Figure 11: *The input BUDDHA mesh has genus 100, the quad mesh obtained with our method has genus 9. All other approaches tested for this paper yield too many singularities to compute a proper parametrization. The DRAGON is meshed at three different target edge lengths and corresponding σ. As the target edge length increases, fewer small features are reproduced. The LION mesh yields a cross field with 1113 singularities when applying [Bommes et al. 2009]. Using our approach we reduce this to 60 singularities and can extract a quad mesh that is not distorted by the bumpy back. The largest bounding box edge lengths and the target edge lengths s used are as indicated. All results were computed using σ = s/2 in accordance with Section 3.*
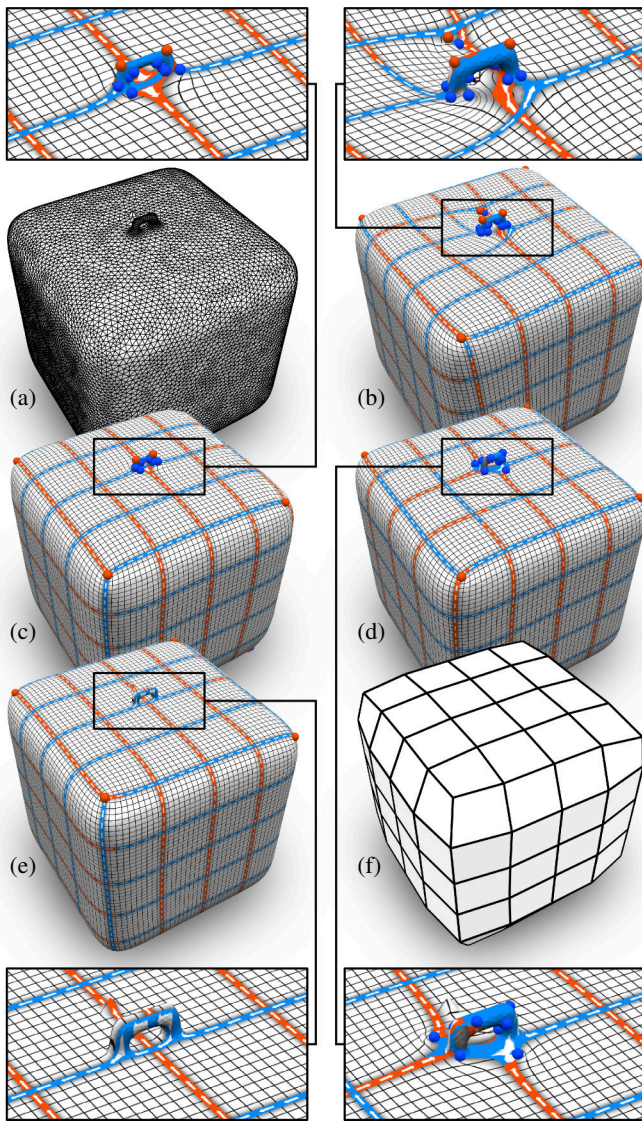
Figure 12: *(a) For a mesh with a small handle and a spike (the same as depicted in Figure 4) a cross field and a parametrization are computed using the method from [Bommes et al. 2009], (b) without additional processing (c) after smoothing the mesh, (d) in conjunction with the method from [Ray et al. 2009] and (e) in conjunction with our approach. Using smoothing, a few pairs of singularities cancel out one another, using [Ray et al. 2009], even more pairs of singularities are cancelled out and the remaining singularities are dispersed. Only when using our approach the mesh is parametrized as if the handle was not existent, allowing the extraction of a coarse quad mesh (f).*

10 Acknowledgments

References

ATTENE, M., CAMPEN, M., AND KOBBELT, L. 2013. Polygon mesh repairing: An application perspective. *ACM Comput. Surv. 45*, 2 (Mar.), 15:1–15:33.

BEN-CHEN, M., GOTSMAN, C., AND BUNIN, G. 2008. Conformal flattening by curvature prescription and metric scaling. *Computer Graphics Forum 27*, 2, 449–458.

BISCHOFF, S., PAVIC, D., AND KOBBELT, L. 2005. Automatic restoration of polygon models. *ACM Trans. Graph. 24*, 4 (Oct.), 1332–1352.

BOMMES, D., ZIMMER, H., AND KOBBELT, L. 2009. Mixed-integer quadrangulation. In *Proc. SIGGRAPH 2009*, ACM, New York, NY, USA, 77:1–77:10.

BOMMES, D., CAMPEN, M., EBKE, H.-C., ALLIEZ, P., AND KOBBELT, L. 2013. Integer-grid maps for reliable quad meshing. In *Proc. SIGGRAPH 2013*, ACM, New York, NY, USA, 98:1–98:12.

BOMMES, D., LÉVY, B., PIETRONI, N., PUPPO, E., SILVA, C., TARINI, M., AND ZORIN, D. 2013. Quad-Mesh Generation and Processing: A Survey. *Computer Graphics Forum 32*, 6, 51–76.

CAMPEN, M., AND KOBBELT, L. 2014. Interactive design of quad layouts using elastica strips. In *Proc. SIGGRAPH Asia 2014*, ACM, New York, NY, USA.

CAMPEN, M., AND KOBBELT, L. 2014. Quad layout embedding via aligned parameterization. *Computer Graphics Forum*.

CAMPEN, M., BOMMES, D., AND KOBBELT, L. 2012. Dual loops meshing: quality quad layouts on manifolds. *ACM Trans. Graph. 31*, 4 (July), 110:1–110:11.

COHEN-STEINER, D., AND MORVAN, J.-M. 2003. Restricted delaunay triangulations and normal cycle. In *Proceedings of the nineteenth annual symposium on Computational geometry*, ACM, New York, NY, USA, SCG '03, 312–321.

CRANE, K., DESBRUN, M., AND SCHRÖDER, P. 2010. Trivial connections on discrete surfaces. *Computer Graphics Forum (SGP) 29*, 5, 1525–1533.

CRANE, K., PINKALL, U., AND SCHRÖDER, P. 2013. Robust fairing via conformal curvature flow. *ACM Trans. Graph. 32*, 4 (July), 61:1–61:10.

DESBRUN, M., MEYER, M., SCHRÖDER, P., AND BARR, A. H. 1999. Implicit fairing of irregular meshes using diffusion and curvature flow. In *Proc. SIGGRAPH '99*, 317–324.

DIAMANTI, O., VAXMAN, A., PANOZZO, D., AND SORKINE-HORNUNG, O. 2014. Designing N-PolyVector fields with complex polynomials. *Computer Graphics Forum 33*, 5, 1–11.

EBKE, H.-C., BOMMES, D., CAMPEN, M., AND KOBBELT, L. 2013. QEx: Robust quad mesh extraction. In *Proc. SIGGRAPH Asia 2013*, ACM, New York, NY, USA, 168:1–168:10.

EL-SANA, J., AND VARSHNEY, A. 1997. Controlled simplification of genus for polygonal models. In *Visualization '97., Proceedings*, 403–410.

GUSKOV, I., AND WOOD, Z. 2001. Topological noise removal. *2001 Graphics Interface Proceedings: Ottawa, Canada* (June), 19–26.

HERTZMANN, A., AND ZORIN, D. 2000. Illustrating smooth surfaces. In *Proc. SIGGRAPH '00*, 517–526.

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

KOVACS, D., MYLES, A., AND ZORIN, D. 2011. Anisotropic quadrangulation. *Computer Aided Geometric Design 28*, 8 (Nov.).

KÄLBERER, F., NIESER, M., AND POLTHIER, K. 2007. Quadcover - surface parameterization using branched coverings. *Computer Graphics Forum 26*, 3, 375–384.

LAI, Y.-K., JIN, M., XIE, X., HE, Y., PALACIOS, J., ZHANG, E., HU, S.-M., AND GU, X. 2010. Metric-driven RoSy field design and remeshing. *IEEE Trans. Vis. Comput. Graph.*, 95–108.

LI, W.-C., VALLET, B., RAY, N., AND LÉVY, B. 2006. Representing higher-order singularities in vector fields on piecewise linear surfaces. *Visualization and Computer Graphics, IEEE Transactions on 12*, 5 (sept.-oct.), 1315 –1322.

LI, E., LÉVY, B., ZHANG, X., CHE, W., DONG, W., AND PAUL, J.-C. 2011. Meshless quadrangulation by global parameterization. *Computers & Graphics 35*, 5, 992–1000.

LIPMAN, Y. 2012. Bounded distortion mapping spaces for triangular meshes. *ACM Trans. Graph. 31*, 4 (July), 108:1–108:13.

MEEK, D., AND WALTON, D. 2000. On surface normal and gaussian curvature approximations given data sampled from a smooth surface. *Computer Aided Geometric Design 17*, 6, 521 – 543.

MYLES, A., AND ZORIN, D. 2012. Global parametrization by incremental flattening. In *Proc. SIGGRAPH 2012*, ACM, New York, NY, USA, 109:1–109:11.

MYLES, A., PIETRONI, N., KOVACS, D., AND ZORIN, D. 2010. Feature-aligned T-meshes. In *Proc. SIGGRAPH 2010*, ACM, New York, NY, USA, 117:1–117:11.

MYLES, A., PIETRONI, N., AND ZORIN, D. 2014. Robust field-aligned global parametrization. In *Proc. SIGGRAPH 2014*, ACM, New York, NY, USA, 135:1–135:14.

PALACIOS, J., AND ZHANG, E. 2007. Rotational symmetry field design on surfaces. *ACM Trans. Graph. 26*, 3, 55.

PANOZZO, D., LIPMAN, Y., PUPPO, E., AND ZORIN, D. 2012. Fields on symmetric surfaces. *ACM Trans. Graph. 31*, 4 (July), 111:1–111:12.

PANOZZO, D., PUPPO, E., TARINI, M., AND SORKINE-HORNUNG, O. 2014. Frame fields: Anisotropic and non-orthogonal cross fields. In *Proc. SIGGRAPH 2014*, ACM, New York, NY, USA, 134:1–134:11.

PIETRONI, N., TARINI, M., SORKINE, O., AND ZORIN, D. 2011. Global parametrization of range image sets. *ACM Transactions on Graphics, Proceedings of SIGGRAPH Asia 2011 30*, 6.

RAY, N., LI, W. C., LÉVY, B., SHEFFER, A., AND ALLIEZ, P. 2006. Periodic global parameterization. *ACM Trans. Graph. 25*, 4 (Oct.), 1460–1485.

RAY, N., VALLET, B., LI, W. C., AND LÉVY, B. 2008. N-symmetry direction field design. *ACM Trans. Graph. 27*, 2 (May), 10:1–10:13.

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

SPRINGBORN, B., SCHRÖDER, P., AND PINKALL, U. 2008. Conformal equivalence of triangle meshes. *ACM Trans. Graph. 27* (August), 77:1–77:11.

TARINI, M., PUPPO, E., PANOZZO, D., PIETRONI, N., AND CIGNONI, P. 2011. Simple quad domains for field aligned mesh parametrization. In *Proc. SIGGRAPH Asia 2011*, ACM, New York, NY, USA, 142:1–142:12.

TAUBIN, G. 1995. A signal processing approach to fair surface design. In *In Proc. SIGGRAPH '95*, ACM, New York, NY, USA, 351–358.

UMENHOFFER, T., SZÉCSI, L., AND SZIRMAY-KALOS, L. 2011. Hatching for motion picture production. *Computer Graphics Forum 30*, 2, 533–542.

ZHOU, Q.-Y., JU, T., AND HU, S.-M. 2007. Topology repair of solid models using skeletons. *Visualization and Computer Graphics, IEEE Transactions on 13*, 4 (July), 675–685.