Approximate Conformal Parameterization of Point-Sampled Surfaces

R. Schmidt and K. Singh

Dynamic Graphics Project
University of Toronto, Canada

Abstract

Drawing on recent machine learning work in dimensionality reduction, novel techniques for approximate conformal parameterization of point-set surfaces are introduced. An improved approximation of local tangent-spaces leads to a new method for computing Laplacian weights on point set neighbourhoods. These weights allow for linear minimization of the Dirichlet energy, and a robust one-parameter minimization of the Conformal energy. The latter technique repairs the well-known distortion of the "two-fixed-point" natural conformal parameterization [DMA02], enables free-boundary parameterization with uniform, authalic, and mean-value weights, and supports hybrid weight matrices to improve parameterization robustness.

1. Introduction

Embedding a three-dimensional surface in the plane has many applications in computer graphics. Most work in planar parameterization minimizes energy functions defined over connected, non-overlapping triangles or vertex one-ring neighbourhoods. Such methods can only be applied to polygonal (often triangular) meshes, where the edge topology can be embedded as a planar graph.

Point-Set Surfaces [KB04, GP07] are an alternative to meshes which relax topological requirements. When a surface is represented only by points, a *neighbourhood* is usually defined as some combination of the k-nearest points and a Euclidean ε-ball. The resulting neighbourhood topology generally cannot be embedded as a planar graph. The standard angle, area, and stretch metrics also do not apply, and without mesh topology it is unclear how to define the boundary of a point set. Hence, mesh parameterization methods cannot be directly applied to point-set surfaces.

Many of the applications of parameterization, such as texture mapping, re-meshing, *etc* [FH02], are also useful in the point set domain. However, few applicable parameterization algorithms are available. We describe a new technique which directly parameterizes a point-set, producing results qualitatively similar to conformal mesh parameterizations without the need for an expensive intermediate meshing step.

1.1. Conformal Mesh Parameterization

For an overview of the parameterization literature, we refer the reader to recent surveys [FH05, SPR06]. Several works specifically target *conformal* or *angle-preserving* parameterization. Angle-Based Flattening [SLMB05] is perhaps the most direct approach, attempting to directly preserve face angles. Discretizing the minimal surface area problem results in the *discrete conformal energy* [PP93], which leads to the well-known *cotangent* weights and can be minimized using linear techniques [DMA02, LPRM02, MTAD08]. We consider these techniques in detail in Section 2.

Conformality can also be formulated in terms of circle patterns [KSS06], or as the result of discrete surface Ricci flow, leading to global conformal parameterizations of arbitrary-genus meshes [JKG07]. To reduce distortion Gaussian curvature can be concentrated at *cone singularities*, between which cuts are automatically added [KSS06]. Approaches combining cone singularites and *metric scaling* have resulted in efficient linear techniques [BCMB08], and a discrete formulation of conformal equivalence results in seamless low-distortion maps [SSP08].

The conformal techniques mentioned above involve specific aspects of mesh geometry. In the meshless domain, Floater and Reimers [FR01] computed shape-preserving weights in tangent-plane Delaunay triangulations, then fixed a manually-determined boundary and solved for the inte-

rior point positions. This method has been adapted to spherical parameterization of genus-0 point sets [ZG04]. The PointShop system [ZPKG02] includes a free-boundary technique, where the user interactively specifies constraint points to ensure bijectivity. Schmidt et al. [SGW06] describe a discrete approximation of the log/exp map on point-sampled surfaces, resulting in tangent-space *normal coordinates*. We improve and extend this technique in Section 3.

1.2. Dimensionality Reduction

Dimensionality reduction involves attempting to reconstruct a map from an M-d Euclidean space to an M-d manifold embedded in N-d, given only point samples on the embedded manifold. Planar parameterization is the special case $(M = 2, N = 3)$. Strong assumptions about the map are often made. For example, IsoMap [TdSL00] assumes that the map is globally isometric and attempts to preserve all geodesic distances between points. Although reasonable parameterizations can sometimes result [ZKK02], IsoMap tends to "flatten" bumps rather than unfold them (Fig. 1a). A conformal variant [dST02] assumes that the M-d space is uniformly sampled, conflicting with desirable surface sampling.

Maximum-Variance Unfolding (MVU) [WS04] makes an even stronger isometric assumption, explicitly preserving edge lengths using semi-definite programming. This leads to a result similar to IsoMap, and even origami-like mesh folding with FastMVU, which extends an initial simplified solution to reduce computation time [WSZS07].

Some dimensionality reduction techniques are based on eigenproblems involving graph Laplacians, and so are related to recent spectral techniques in graphics [ZvKD09] and roughly as efficient as sparse linear methods [MTAD08]. In particular, Locally-Linear Embedding (LLE) [RS00] and Laplacian Eigenmaps (LEM) [BN03] only assume *local* isometry, and hence are more applicable to 3D surface parameterization. As we show in Section 2, LEM directly minimizes the Dirichlet energy [PP93]. LLE has been used before in the graphics domain [PC05], where accurate geodesic distances between a sparse set of mesh vertices were used to create an initial LLE map, which was then extended to a full parameterization via geodesics-based interpolation.

Hessian LLE [DG03] is based on the same framework as LLE and LEM, but replaces the local Laplacian operator with a Hessian norm. Direct application of HLLE can give good results for some parameterization problems [LYD*05], but fails in cases involving more distortion (Fig. 1f,h). Another technique, Local Tangent-Space Alignment (LTSA) has also been applied in graphics domains [CLZW07], but appears to generate results very similar to HLLE (Fig. 1e,g)

1.3. Contributions

Local tangent spaces are often approximated using planar projection. The Discrete Exponential Map (DEM) [SGW06] offers a geometry-aware alternative. We improve this method, then use the local tangent spaces to compute graph Laplacian weights "on the manifold", and estimate the point-set boundary. We then solve spectral Dirichlet and conformal parameterization problems by adapting Laplacian Eigenmaps (LEM) [BN03] and Spectral Conformal Parameterization (SCP) [MTAD08]. A schematic overview is shown below. Our approach can also be applied to minimize the conformal energy with other types of weights. We show that on meshes it removes the well-known distortion in the two-point *natural* conformal parameterization (DNCP) [DMA02] and can be used to generate free-boundary maps with uniform, authalic, and mean-value weights. Hybrid weight matrices can also be used to resolve foldover problems caused by negative cotangent weights.

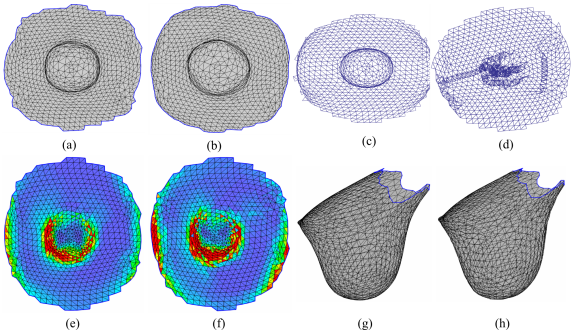
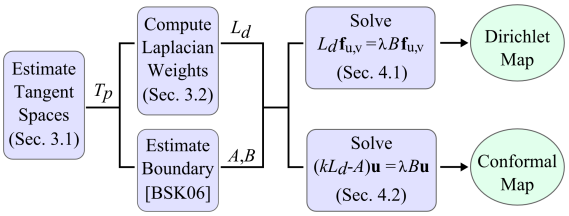


Figure 1: *Left mesh from Fig. 4 flattened using Isomap (a), Conformal Isomap (b), MVU (c) and FastMVU (e). In all cases, assumptions about isometry or isotropy result in undesirable parameterizations. LTSA (e,g) and HLLE (f,h) give good results, but fail in almost exactly the same way on the more complex surface from Fig. 6.*



A short note about our figures. To aid visual comprehension, we will show explanatory results on simple semi-regular triangle meshes, where the vertex set defines an adequately-sampled point-set surface. However, unless otherwise noted, all topological information is discarded before giving the vertex set to the algorithms.

2. Background

In this section we review key aspects of conformal parameterization and dimensionality reduction which will be needed for the following sections.

In the continuous case, parameterization involves finding maps f_u and f_v which take points $\mathbf{p} = (p_x, p_y, p_z)$ on surface $\mathcal{S}$ to coordinates $\mathbf{u} = (u, v)$ in the plane. The *Dirichlet energy* describes one type of desirable map

$$\mathcal{E}_D(f) = \frac{1}{2}\int_{\mathcal{S}} \|\nabla_{\mathcal{S}} f\|^2 \tag{1}$$

If we fix f at the boundary, then this energy can be minimized by finding the *harmonic* that satisfies $\triangle_{\mathcal{S}} f = 0$. Here $\triangle_{\mathcal{S}}$ is the *Laplace-Beltrami* operator, the *Laplace* operator $\triangle = \partial^2_{xx} + \partial^2_{yy}$ "on the manifold".

In the discrete case, $\mathcal{S}$ is generally taken to be an open triangle mesh with V vertices $\{\mathbf{p}_i\}$. For piecewise-linear maps one only need consider the maps at the vertex positions $\mathbf{f} = \{\mathbf{f}_i\}$. In this case, the discrete Dirichlet energy is defined as a sum over the Dirichlet energy at each triangle [PP93]. Written in terms of edges e_{ij} in the mesh,

$$E_D(\mathbf{f}) = \frac{1}{4}\sum_{e_{ij}} \mathrm{W}_{ij}\|\mathbf{f}_i - \mathbf{f}_j\|^2 \tag{2}$$

where the $V \times V$ matrix $\mathrm{L}_D = \mathrm{D} - \mathrm{W}$ is the *graph Laplacian*, a discrete analog to the Laplace-Belatrami operator. The matrix terms W_{ij} are the edge weights, nonzero if $j \in Nbr(i)$, and $\mathrm{D}_{ii} = \sum_j \mathrm{W}_{ij}$. Common weighting schemes include the uniform weights $\mathrm{W}^{\mathrm{uni}}$, cotangent weights $\mathrm{W}^{\mathrm{cot}}$ [PP93], and the mean-value weights [Flo03].

As $E_D(\mathbf{f})$ is quadratic, if we fix the boundary we can find the maps at interior vertices by solving the linear system

$$\mathrm{L}_D\mathbf{f} = 0 \qquad \mathbf{f}_i|_{\partial\mathcal{S}} = \mathbf{f}_i^{\mathcal{B}} \tag{3}$$

The solution is a *discrete harmonic*. Note that Equation 3 is solved separately for $\mathbf{f}_u$ and $\mathbf{f}_v$. The result is guaranteed to be foldover-free with positive weights and limited non-convex boundaries [SJGT06], but in general fixed boundaries will introduce significant distortion (Fig. 4).

LEM The *Laplacian Eigenmaps* technique [BN03] presents an alternative which directly minimizes E_D without any explicit point constraints, by solving the generalized eigenproblem $\mathrm{L}_D\mathbf{f} = \lambda\mathrm{D}\mathbf{f}$. If the eigenvalues are sorted in increasing order $\lambda_0 < \lambda_1 < \ldots$, then the eigenvectors or *eigenfunctions* associated with the eigenvalues are orthogonal minimizers of the *Rayleigh quotient*

$$\left| \mathbf{f}^t \mathrm{L}_D\mathbf{f} \,/\, \mathbf{f}^t \mathrm{D}\mathbf{f} \right| \tag{4}$$

The smallest eigenvalue $\lambda_0 = 0$ corresponds to the degenerate solution which collapses $\mathbf{u}$ to a point. Hence, the parameterization is defined by the next two eigenvalue/eigenvector pairs, $\mathrm{L}_D\mathbf{f}_u = \lambda_1\mathrm{D}\mathbf{f}_u$ and $\mathrm{L}_D\mathbf{f}_v = \lambda_2\mathrm{D}\mathbf{f}_v$.

Locally-Linear Embedding (LLE) [RS00] takes the same approach but minimizes the energy $\mathbf{f}^t\mathrm{L}_D^t\mathrm{L}_D\mathbf{f}$, which approximates the iterated Laplacian $\triangle^2_{\mathcal{S}} f$ [BN03]. This form is symmetric positive-definite even if W is non-symmetric, for example if the rows are normalized.

DNCP An alternative to directly minimizing the Dirichlet energy is to consider the *conformal energy*

$$\mathcal{E}_C(\mathbf{u}) = \mathcal{E}_D(\mathbf{u}) - \mathcal{A}(\mathbf{u}) \tag{5}$$

where $\mathcal{E}_D(\mathbf{u}) = \mathcal{E}_D(\mathbf{f}_u) + \mathcal{E}_D(\mathbf{f}_v)$ and $\mathcal{A}(\mathbf{u})$ is the area in the plane. It can be shown that $\mathcal{E}_D > \mathcal{A}$, so $\mathcal{E}_C \geq 0$, with equality only when $\mathbf{u}$ is *conformal*, meaning that it can be flattened without any distortion in angles [PP93, DMA02].

In the discrete case, the area of a polygon can be computed by summing along the boundary edges e_{ij}

$$\mathcal{A}(\mathbf{u}) = \frac{1}{2}\sum_{e_{ij}\in\partial U} (u_i v_j - u_j v_i) \tag{6}$$

This formula couples the u and v dimensions. Writing $\mathbf{u}$ as the $2V \times 1$ vector $[\mathbf{f}_u, \mathbf{f}_v]$, we then have $\mathcal{A}(\mathbf{u}) = \frac{1}{2}\mathbf{u}^t\mathrm{A}\mathbf{u}$, where A is the $2V \times 2V$ matrix defined in Mullen et al. [MTAD08]. The graph Laplacian is rewritten $\mathrm{L}_D \leftarrow [\mathrm{L}_D, \mathbf{0}; \mathbf{0}, \mathrm{L}_D]$ and then the discrete conformal energy is

$$E_C(\mathbf{u}) = E_D(\mathbf{u}) - \mathcal{A}(\mathbf{u}) = \frac{1}{4}\mathbf{u}^t(\mathrm{L}_D - \mathrm{A})\mathbf{u} = \mathbf{u}^t\mathrm{L}_C\mathbf{u} \tag{7}$$

The A matrix provides Neumann (derivative) conditions on the boundary, so only two vertices need be specified to fix the 2D translation, rotation, and scale [DMA02].

SCP The choice of which vertices to fix can significantly affect the quality of DNCP. Mullen et al. [MTAD08] take the same approach as LEM, solving the eigenproblem $\mathrm{L}_C\mathbf{u} = \lambda\mathbf{u}$. Note that in this case only one eigenvector is necessary. Mullen et al. also observe that the denominator of the minimized Rayleigh quotient (Eq. 4) balances discrete conformality against the sum of squared distances, implying that each vertex is essentially connected to a spring pulling it towards the origin. This can result in highly distorted foldover regions. Hence, they propose a modified eigenproblem $\mathrm{L}_C\mathbf{u} = \lambda\mathrm{B}\mathbf{u}$, where $\mathrm{B}_{ii} = 1$ if $\mathbf{p}_i$ is a boundary vertex. In this case the springs are only attached to boundary vertices; the interior vertices are otherwise unconstrained.

3. Local Weights on Point Sets

Minimizing the Dirichlet or conformal energy on a point set requires a suitable graph Laplacian. We first approximate the tangent-space at each point by improving the Discrete Exponential Map (DEM) [SGW06]. The result provides a local isometry in which we can compute "optimal" local weights.

The DEM approximates the exp / log map at a point $\mathbf{s}$, mapping the local neighbourhood into the *tangent-space* $T_{\mathbf{s}}$ via a propagation of *normal coordinates* outwards from $\mathbf{s}$ [SGW06]. Given a piecewise-linear path $\{\mathbf{s}, \mathbf{q}_1, \ldots, \mathbf{q}_n, \mathbf{p}\}$ from $\mathbf{q}$ to each point $\mathbf{p}$. The tangent-space coordinates $T_{\mathbf{s}}\mathbf{p}$ are then defined recursively as

$$T_{\mathbf{s}}\mathbf{p} = T_{\mathbf{s}}\mathbf{q} + M_{\mathbf{qs}}T_{\mathbf{q}}\mathbf{p} \tag{8}$$

where $T_{\mathbf{q}}\mathbf{p}$ is found via length-preserving tangent-plane projection of $(\mathbf{p} - \mathbf{q})$, and $M_{\mathbf{qs}}$ is a 3D rotation aligning the

tangent-normal frame $\mathcal{F}_q$ with $\mathcal{F}_s$ [SGW06]. If an approximate geodesic front propagation is used to compute the paths, for example via Dijkstra's algorithm on the neighbourhood graph, then Eq. 8 can be evaluated in-line and runtime complexity is unaffected.

3.1. Upwind-Average Discrete Exponential Map

One limitation of the DEM is that any error introduced at $\mathbf{q}_i$ will be propagated to downwind points whose path passes through $\mathbf{q}_i$, potentially leading to catastrophic failures (Fig. 3b). Similarly, as the paths are completely independent the error can vary wildly between two neighbouring points. Since the DEM sums vectors rather than scalars, $T_\mathbf{s}\mathbf{p}$ could be estimated from *any* nearby upwind point (Fig. 2). The result will be slighly different in each case, so we re-define $T_\mathbf{s}\mathbf{p}$ as a weighted average of several estimates:

$$T_\mathbf{s}\mathbf{p} = \sum_i w(\mathbf{p},\mathbf{q}_i)\,(T_\mathbf{s}\mathbf{q}_i + M_{\mathbf{q}_i\mathbf{s}} T_{\mathbf{q}_i}\mathbf{p})) \qquad (9)$$

where $\mathbf{q}_i$ are nearby upwind neighbours to $\mathbf{p}$ (Figure 2) and w is the inverse distance weight $w(\mathbf{p},\mathbf{q}) = (\|\mathbf{p}-\mathbf{q}\|^2+\varepsilon)^{-1}$. As shown in Fig. 3, upwind averaging improves DEM robustness, with a small 5-10% increase in runtime cost.

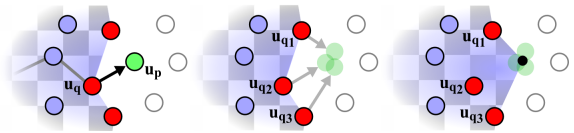


Figure 2: *Although the Discrete Exponential Map estimates a uv-parameter $\mathbf{u}_p$ from a single upwind sample $\mathbf{u}_q$ (a), other nearby points on the uv-front provide equally likely estimates (b) which can be averaged to enhance DEM robustness (c).*

As the DEM uses normal information, smoothing normals relaxes the parameterization in regions of higher curvature. The result is essentially the parameterization of a smoother surface. Replacing each normal with a distance-weighted average of k-neighbourhood normals results in a significant improvement and can be efficiently evaluated in-line with the DEM. Combined with upwind averaging, normal smoothing results in much more stable maps (Figure 3).

3.2. Tangent-Space Optimal Weights

To find neighbourhood weights, LLE [RS00] explicitly minimizes the *reconstruction error*

$$R_\mathbf{x} = \|\mathbf{x} - \sum w_j \mathbf{x}_j\|^2 \qquad (10)$$

subject to the constraint that $\sum w_j = 1$. The solution $\mathbf{w} = \{w_j\}$ can be found by solving the linear system

$$\mathrm{C}\mathbf{w} = 1 \qquad \mathrm{C}_{jk} = (\mathbf{x}-\mathbf{x}_j)\cdot(\mathbf{x}-\mathbf{x}_k) \qquad (11)$$

where C is the Gram (covariance) matrix, and then normalizing $\mathbf{w}$. C is ill-conditioned when the number of neighbours is

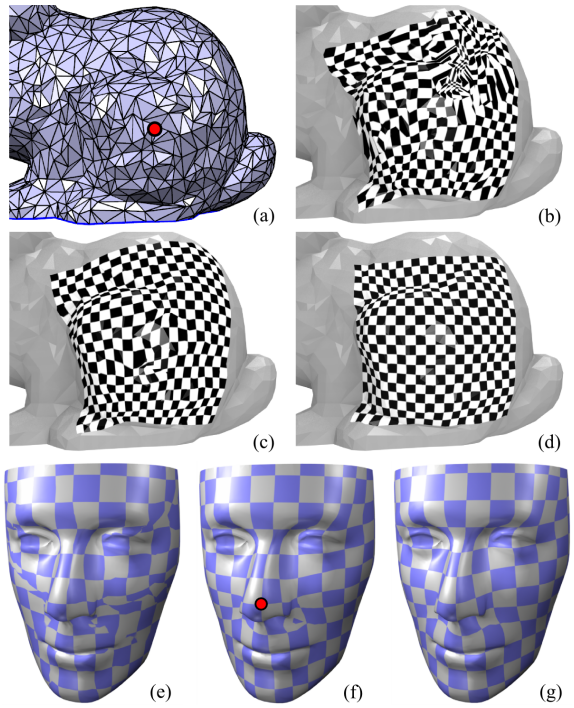


Figure 3: *For certain points (red dot) on a highly irregular bunny mesh (a), the original DEM fails catastrophically (b). The upwind-average DEM is more robust (c), and with the addition of local normal smoothing (d) a low-distortion parameterization is produced. (e-g) shows the same progression over a surface with wide variation in curvature, which is also problematic for the DEM.*

greater than the dimension (nearly guaranteed for $\mathbb{R}^3 \to \mathbb{R}^2$), and so must be regularized by adding a small multiple ε of the identity matrix:

$$\mathrm{C} \leftarrow \mathrm{C} + (\varepsilon \mathrm{Tr}(\mathrm{C}))I \qquad (12)$$

The techniques described in Sec. 2 attempt to preserve the relationships between each point and its neighbours, as expressed by the weights. The implicit assumption in minimizing $R_\mathbf{x}$ in 3D is that it will approximate minimal $R_\mathbf{x}$ in 2D. At points with any appreciable curvature, however, $\mathbf{x}$ will lie outside the convex hull of its neighbourhood, so minimizing Eq. 10 can produce large negative weights. The flattened neighbourhoods which best match these weights may not be desirable or even geometrically consistent.

Intuitively, the ideal weights would minimize reconstruction error in the parameterization. This can be approximated by explicitly flattening the region around $\mathbf{x}$ before computing weights. Such weights will at minimum describe a geometrically plausible neighbourhood, albeit one which reflects the properties of the flattening technique. As it preserves some intrinsic geometric properties of the surface, the tangent-

space $T_{\mathbf{x}}$ is a suitable candidate. This space is linear, and geodesic distances from $\mathbf{x}$ to its neighbours are preserved, so from the point-of-view of $\mathbf{x}$ the tangent map is an isometry. Using the approximate tangent-space coordinates generated via Eq. 9, the weights problem can be rewritten as

$$\arg\min_{\mathbf{w}} \|\mathbf{0} - \sum w_j T_{\mathbf{x}} \mathbf{x}_j\|^2 \qquad (13)$$

These tangent-space weights $\mathrm{W}^{\mathrm{tan}}$ significantly improve LLE (Fig. 4). As the DEM is $O(k \log k)$ for a k-neighbourhood, and solving Eq. 13 is $O(k^2)$, the asymptotic complexity of most algorithms is not affected, although there is a noticeable computation cost (Fig. 5).

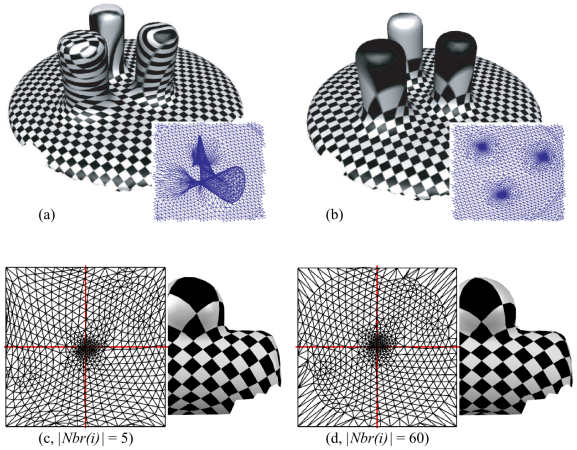
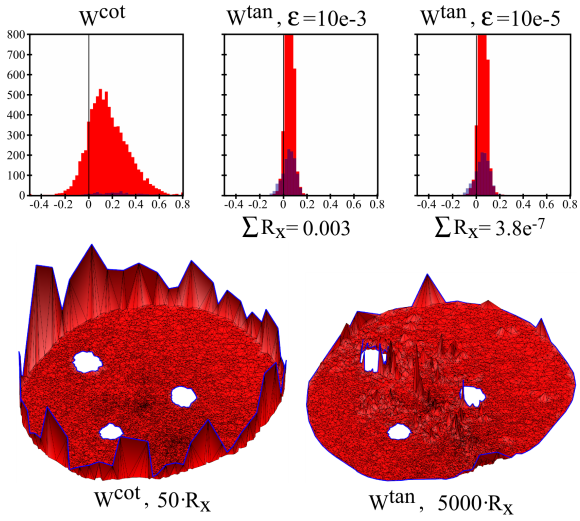


Figure 4: *LLE with 3D (a) and tangent-space (b) weights, which allow large neighbourhoods to be used, reducing interior distortion in fixed-boundary parameterizations (c,d).*

One advantage of computing weights in a flattened neighbourhood is that we are not restricted to using small "locally linear" portions of the surface. We have found that increasing the overlap between neighbourhoods results in a "stiffer" deformation. This has a particularly noticeable effect when fixing boundary vertices to a convex polygon and solving Eq. 3. As the neighbourhood size increases, the interior pulls away from the polygon and approaches a free-boundary result (Fig. 4). For free-boundary methods, larger neighbourhoods mainly increase robustness to undersampling.

The other free parameter is the regularization tolerance ε. While Eq. 13 does not explicitly guarantee positive weights, we observe fewer negative weights at interior points as ε increases, although the reconstruction error $R_{\mathbf{x}}$ also grows. Using too large or too small a value will reduce smoothness in the map. Empirically we find that $\varepsilon = 10e^{-3}$ gives results similar to $\mathrm{W}^{\mathrm{cot}}$ with low $R_{\mathbf{x}}$, and hence is nearly planar-reproducing. We note that there are fewer negative weights on the interior than $\mathrm{W}^{\mathrm{cot}}$, but more on the boundary. This is because $\mathrm{W}^{\mathrm{tan}}$ still minimizes $R_{\mathbf{x}}$ on the boundary, which is not the case for $\mathrm{W}^{\mathrm{cot}}$ (Fig. 5).



	|Ni| ~ 15	|Ni| ~ 30	|Ni| ~ 60
DEM/sec	22.5k	13.6k	7.4k
Weights/sec	4.4k	1.3k	0.4k

Figure 5: *Weight value histograms for interior (red) and boundary (blue) points on an irregular bunny mesh, comparing W^{cot} and W^{tan} with several values of ε. Heights in second row are scaled reconstruction errors $R_{\mathbf{x}}$. Table breaks down computation time for varying neighbourhood sizes into tangent-space estimation/s and solves of Eq. 13/s.*

4. Point-Set Parameterization

As $\mathcal{E}_D \geq \mathcal{A}$, minimizing the Dirichlet energy approximates a conformal parameterization. We present a linear method for minimizing E_D using our tangent-space weights $\mathrm{W}^{\mathrm{tan}}$. We find that this method, called *Spectral Dirichlet Parameterization* or **SDP**, is effective in cases with low to moderate distortion. To minimize the more robust conformal energy, we must find a scaling factor to compensate for the fact that $\mathrm{W}^{\mathrm{tan}}$ is row-normalized. We refer to this technique as *Optimized Conformal Parameterization*, or **OCP**.

4.1. Spectral Dirichlet Parameterization (SDP)

We first consider whether the row-normalized $\mathrm{W}^{\mathrm{tan}}$ still approximates a Dirichlet energy. In the parameterization literature, the L_D and E_D have been presented as discrete analogs of $\triangle_{\mathcal{S}}$ and $\mathcal{E}_D$. The "correct" weights $\mathrm{W}^{\mathrm{cot}}$ can then be derived from the Dirichlet energy of piecewise-linear functions on mesh triangles [PP93]. No such discretization currently exists for point sets. However, it can be shown [BN03] that in a locally-linear patch around a point $\mathbf{p}_i$,

$$[(I - W)f]_i \approx \frac{1}{2} k_i \triangle_{\mathcal{S}} f \qquad (14)$$

where k_i is a proportionality constant. Hence, L_D is an *approximation* to $\triangle_S$ on an underlying smooth manifold, so with our approach it is still the case that $E_D \approx \mathcal{E}_D$.

LEM minimizes the Dirichlet energy without the need for any explicit boundary constraints. However, for cases where any appreciable distortion is necessary, LEM fails catastrophically (Fig. 6b). As was found in Mullen et al. [MTAD08], the issue here is that the Rayleigh quotient being minimized (Eq. 4) prefers a specific global point distribution. Hence, as in **SCP** we solve the generalized eigenproblem $L_D\mathbf{f} = \lambda \mathrm{B}\mathbf{f}$, where B is the matrix with 1's on the diagonal at boundary points. This leads to foldover-free LEM parameterizations (Fig. 6c), even in highly distorted cases.

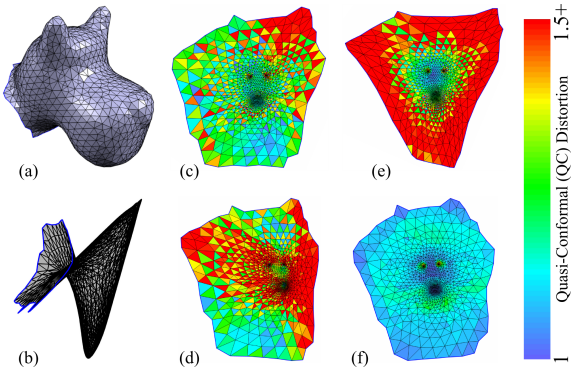


Figure 6: *A surface (a) parameterized using LEM with W^{tan} (b), the effect of adding the B matrix to LEM (c) and LLE (d), LEM with W^{cot} (e), and **SCP** (f).*

One challenge with meshless approaches is that without topology, deciding which are "boundary" points is nontrivial. Intuitively, a point is on the boundary of a manifold when its local neighbourhood only covers a portion of the tangent-space. Hence, boundary points can be detected by projecting neighbours into the tangent-space and looking for large gaps in the spoke angles around the origin. This is known as the *angle criterion* [BSK06]. On well-sampled point-set surfaces, we have found that using the angle criterion in our estimated tangent spaces finds a suitable set of boundary points (Fig. 7). Even if we decimate the boundary set by half, the parameterization is only slightly altered.

Finally, we mention that LLE can be substituted for LEM in the above discussion. We can even formulate a conformal variant of LLE, using the methods of the next section. However, despite being positive-definite and hence numerically more desirable, we found found that LLE parameterizations are at best similar to LEM, and often have higher distortion (Fig. 6d), so we will not consider LLE any further.

4.2. Optimized Conformal Parameterization (OCP)

One limitation of the spectral Dirichlet parameterization is that, as each dimension is independent, they may have different scales. This can lead to a shearing effect in the parameterization (Fig. 8b). The conformal energy couples the two planar dimensions (Eq. 5). Solving SCP with $\mathrm{W}^{\mathrm{tan}}$ produces what appears to be catastrophic failure, but on closer examination we see that most of the mesh has been reasonably parameterized (Fig. 8d), and both $\mathcal{A}(\mathbf{u})$ and the area of each triangle remains positive.

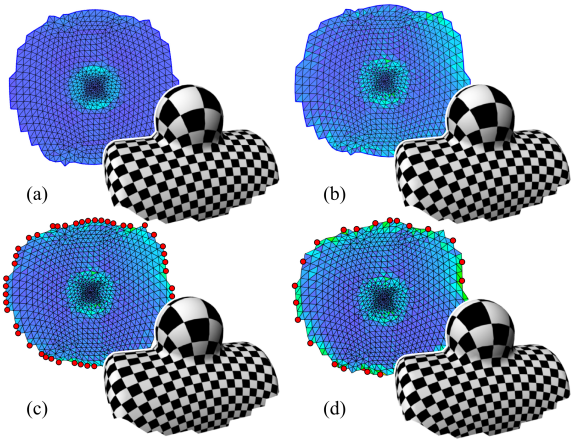


Figure 7: *The bump mesh parameterized with W^{cot}-SCP (a), W^{tan}-LEM with the mesh boundary vertices (b), estimated boundary points (c), and 50% of the estimated boundary points (d).*

We observe that the conformal energy (Eq. 7) can be interpreted as a sum of quadratic terms, which attempts to balance the Dirichlet energy and the area. However, our weight matrix $\mathrm{W}^{\mathrm{tan}}$ is normalized. If we compute SCP with normalized $\mathrm{W}^{\mathrm{cot}}$, we see a similar effect, suggesting that the result is dependent on the *scale* of each row of L_D. The same scaling factor is found in Eq. 14. Unlike for $\mathrm{W}^{\mathrm{cot}}$, we do not know this scale a priori, and so will have to find it. As the necessary V-dimensional optimization would be to expensive, we formulate a single-variable approxmiation

$$\widehat{E_C} = (\mathbf{u}^t(k\mathrm{L}_D - \mathrm{A})\mathbf{u})^2 \qquad (15)$$

We then minimize the energy $\widehat{E_C}^2$ by applying a line-search to k, where at each evaluation we solve the resulting quadratic problem using SCP once k is fixed.

As k moves away from the optimal value k^{opt}, one might expect $\widehat{Dc}$ to simply increase. However, $\mathcal{A}(\mathbf{u})$ does not measure the point-set area of the parameterization, but rather the *winding area*. Consider a circle with circumference c. If we deform this boundary to create two overlapping circles with circumference $c/2$, the total area drops by a factor of 4. Hence, as k decreases additional low-energy states can be found where the *turning number* of the boundary polygon is greater than 1. This explains why $\mathcal{A}(\mathbf{u})$ remains positive: the parameterization has not folded, instead it has "spun".

This spinning is clearly visible when observing the pa-

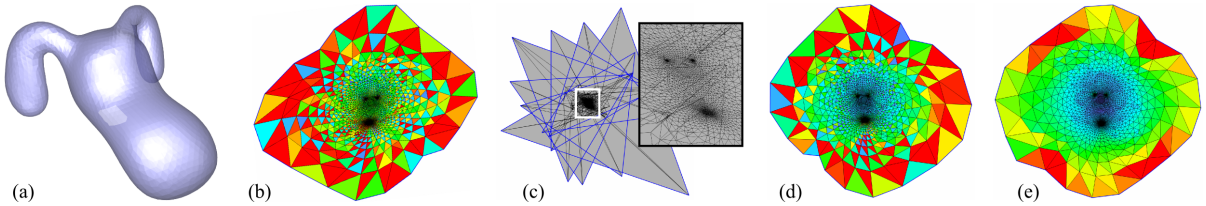


Figure 8: *In this example a mesh with several extended protrusions is flattened into a small hole in it's base (a). LEM fails to accurately reproduce the relationship between the two parameterization axes (b). Solving SCP using W^{tan} appears to catastrophically fail (c), but looking closely we find that most of the parameterization is similar to (b). After finding the optimal scaling factor k^{opt} for W^{tan}, we achieve a result (d) similar to SCP with W^{cot} (e).*

rameterization at different points on the energy landscape generated by varying k (Fig. 9). We see that $\widehat{Dc}$ is highly nonlinear, but has a regular structure that makes the desired critical point easy to find. Essentially, the turning number $t(\mathbf{u})$ of the boundary polygon decreases as $k \to k^{\text{opt}}$ from below, while once $k > k^{\text{opt}}$, $t(\mathbf{u}) = 1$ and $\widehat{Dc}$ increases to infinity. Hence, we can first search forward from $k = 1$ until we find the $t(\mathbf{u}) = 1$ portion of the landscape, and then minimize Eq. 7. Within each $t(\mathbf{u})$ step, we essentially have a quadratic bowl, which makes optimization very efficient. Even with a gradient-free method like Matlab's *fminbnd*, we converge in 10-15 iterations. In some cases the minima can be very closely spaced, and the optimizer may skip back and find a lower value in one of the previous bowls. To avoid this we define a modified energy

$$\widehat{E_{ct}} = (\mathbf{u}^t(k\mathrm{L}_D - \mathrm{A})\mathbf{u})^2 + r|t(\mathbf{u})| \qquad (16)$$

where $r \gg \min \widehat{E_{ct}}$ (we use $r = 10$). This shifts the "bowls" in Fig. 9 upwards at each discrete step of $t(\mathbf{u})$, preventing the optimizer from finding lower minima with $t(\mathbf{u}) > 1$.

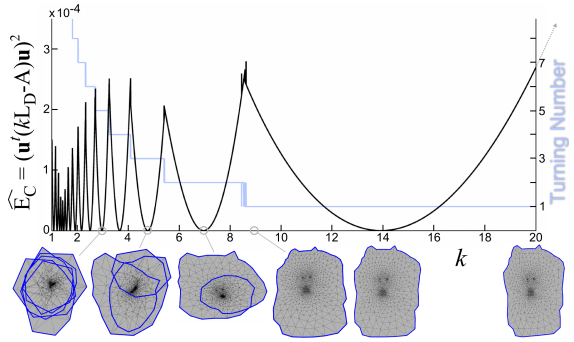


Figure 9: $\widehat{E_C}$ *landscape while varying k.*

Since W^{tan} is non-symmetric, some eigenvalues may have a complex component. However, note that if we rewrite our 2D coordinates as complex numbers $u + iv$, the $2V \times 2V$ real matrix L_C can be rewritten as a $V \times V$ matrix with complex entries, and we find the same solution. Alternately, since W^{tan} is real, the complex conjugate of any complex eigenvalue is also an eigenvalue, and the corresponding eigenvectors are also complex conjugates. Since both minimize $\mathcal{E}_C$ where $\mathbf{u}$ is a $2V \times 1$ complex vector, the projection onto the real line also minimizes $\mathcal{E}_C$.

5. Optimized DNCP

It is interesting to consider the effect of our optimization procedure on mesh parameterization. While we do not improve W^{cot}-SCP, it is well-known that the conformal distortion of DNCP varies depending on which boundary points are chosen, and that SCP does not suffer from this problem [MTAD08]. If we use DNCP to solve for $\mathbf{u}$ in Equation 16, we find that at k^{opt} the parameterization is visually indistinguishable from the SCP result (Fig. 10).

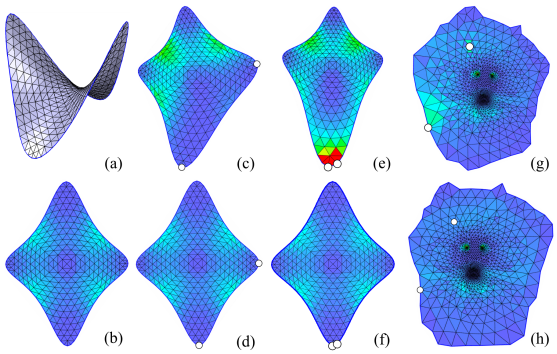


Figure 10: *The symmetry of a saddle mesh (a) is preserved by SCP (b). The constraint points needed for DNCP introduce variable distortion (c,e,g) but applying our optimization procedure reproduces the SCP result, regardless of where the constraints are placed (d,f,h).*

It is also interesting to consider the use of other weights in DNCP. For example, consider the *uniform* weights, W^{uni}. At interior vertices the uniform weights are simply scaled cotangent weights on regular polygons. Hence, W^{uni} describes a hypothetical mesh in which each one-ring is a regular polygon. Plugging W^{uni} into our optimized DNCP, we find a free-boundary parameterization (Fig. 11).

For both W^{uni} and W^{cot}, k^{opt} is very close to 1. If we

normalize each row, we find quite different results in both cases. Essentially, normalizing the rows corresponds to giving each vertex equal weight, while in the canonical forms W^{uni} and W^{cot} generally place a lower weight on boundary vertices. So in some sense, these matrices implicitly contain "local stiffening" factors [BZK09]. We can also solve optimized DNCP and SCP using the area-preserving *authalic* weights [DMA02], and the mean-value weights [Flo03], although the results are not particularly desirable (Fig. 11).

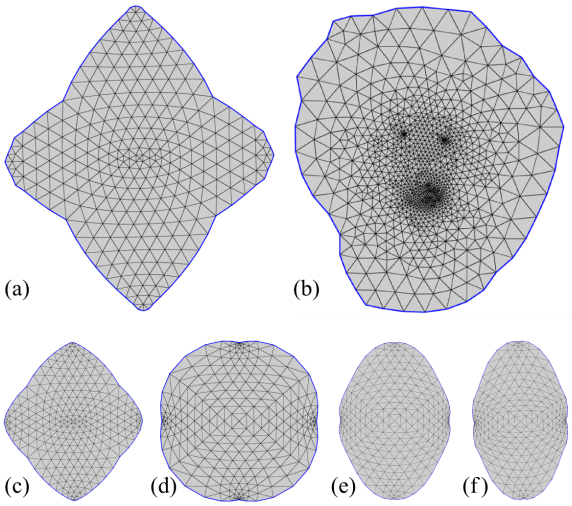


Figure 11: *Optimized DNCP with uniform one-ring weights results in highly regular planar triangulations (a,b). Normalizing the uniform (c) and cotangent (d) weights implicitly increases the relative weight of the boundary vertices. A similar effect is seen with authalic (e) and mean-value weights (f). Saddle mesh is used in (a,c-f), and Fig.6a in (b).*

One consequence our approach is that, as we have much more flexibility in terms of specifying a weight matrix, it is no longer necessary to use the same weight technique at each vertex. This can improve the robustness of existing algorithms. For example, it is well-known that the cotangent weights can be negative. On the poorly-tessellated bunny mesh in Fig. 12 there are many negative weights at the base of the ears, leading to large foldovers. However, if we replace the cotan weights with uniform weights on the ear vertices and solve for t^{opt}, we resolve the foldovers while leaving the rest of the parameterization largely unchanged.

6. Results

The examples we have considered thus far are somewhat artificial, in that they ultimately came from meshes. We now consider several point-sets from PointShop3D [ZPKG02], for which mesh topology was not available.

We have implemented our techniques in Matlab, with the DEM computation done using C++ code. Our software and

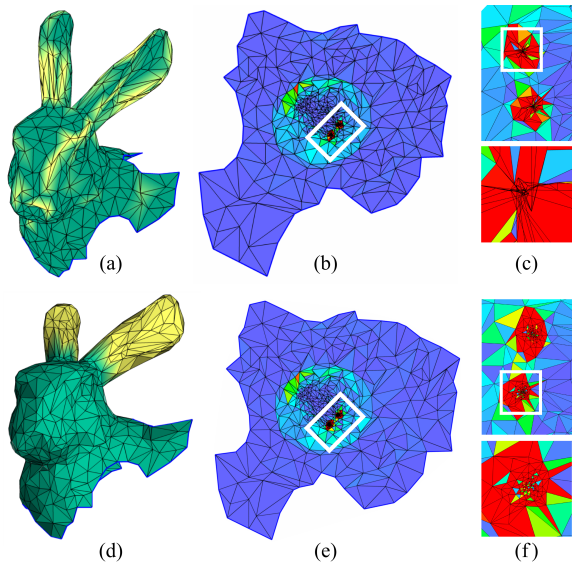


Figure 12: *A highly irregular bunny mesh (a) has many interior vertices with negative cotan weights (yellow). The resulting parameterization (b) has extensive foldovers in the ear region (c). We construct a hybrid weight matrix by replacing the cotan weights with uniform weights on the ear vertices (d, yellow), and solving for optimized SCP. The result is visually similar (e), but the foldovers are avoided (f).*

results are available from (*witheld for anonymous submission*). To run our tests, we first constructed a Euclidean K-NN neighbourhood graph with $K = 8$. We then compute the average edge-length l, and approximate T_p at each point. Within each tangent-space, we construct a geodesic K-ε ball with $K = 6$ and $\varepsilon = 2.5l$, then estimate the boundary loop, compute W^{tan}, and solve for the parameterization. Optionally, we compute a Delaunay triangulation in the plane. Assuming the map has no foldovers, this is a trivial way to mesh a point-set surface, and in near-conformal parameterizations the resulting mesh is of very high quality.

Our first test is the *Gnome* model (Fig. 13). This surface is nearly-symmetric and mainly regularly sampled, with a small strip of irregular sampling where it appears that two scans were marged. We see that the Dirichlet parameterization (**SDP**) preserves this symmetry but has high quasi-conformal (QC) distortion [KSS06], while the conformal map (**OCP**) has low QC-distortion but some symmetry is lost. We have found this to be a general trend - in this case and most others, the boundary loop is not symmetric, which has a more significant effect in **OCP**.

Using the near-regular parameter-space Delaunay mesh, we can compare to **SCP**. We find very low QC-distortion but the irregular region causes significant asymmetry. Using the area-weighted variant of **SCP** described in Mullen et al. [MTAD08] improves symmetry but increases QC-

distortion. This example shows that our tangent-space weights confer some independence from sampling regularity, suggesting that there may be advantages to a meshless approach even if a mesh is available.

Note, however, that while our techniques can handle irregular sampling, they do require a point set dense enough to adequately represent the surface. While this is generally the case for point-set models, one advantange of a meshless approach is that a point set can be trivially subsampled by discarding points. It is then straightforward to compute the DEM on the reduced set for each remaining point, and embed it using barycentric interpolation. However, as we see in Fig. 14, too sparse a sampling leads to high QC-distortion and a "sheared" mesh.

Finally, as with **SCP** our method can handle surfaces with holes (Fig. 15). In this case and the previous example, the subsampled point-set is dense but highly irregular. This is not an issue for our weight computation, but is problematic for our naive boundary estimator. In both Figures 14 and 15 we observe a small "pointy" region of high QC-distortion. On closer examination we find that this region is has an 'outlier' sample. Even on a flattened version of the mesh, we find very high reconstruction error (Fig. 5) and large negative weights at this point. The same effect occurs on a smaller scale at other dark red boundary regions in each QC-distortion image. Hence, more robust boundary estimation and outlier removal will improve these results.

We also experimented with parameterizing raw scan data,

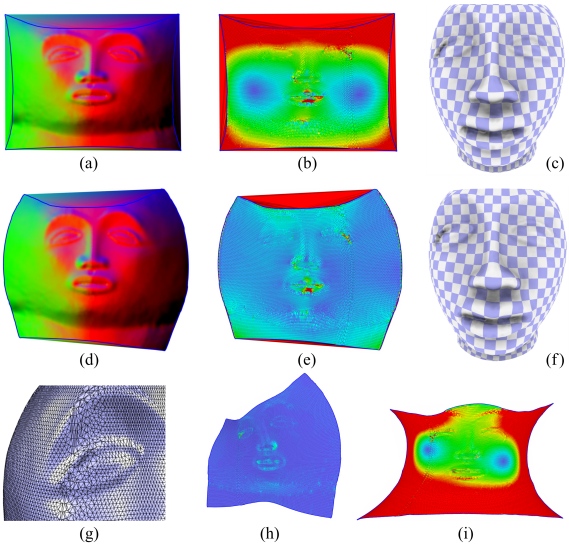


Figure 13: *Gnome model (13k pts) flattened with **SDP** (top row) and **OCP** (middle row). Left column shows quasi-conformal (QC) distortion. A strip of irregular samples (g) leads to a loss of symmetry in **SCP** (h), while the area-weighted **SCP** variant has high QC-distortion (i).*

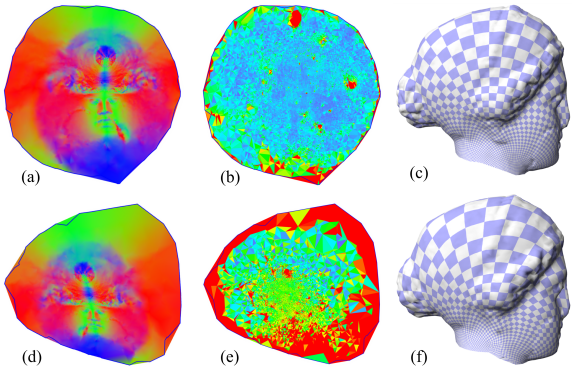


Figure 14: *Igea model (130k pts) with **OCP** at 1/4 (top) and 1/10 (bottom) resolution.*

but found two problems. First, raw scans are rarely manifold, and second, scans are usually very sparse and noisy around boundaries, which is problematic since we depend on good boundaries. Although our methods could perhaps be made robust to these issues, it seems more practical to rely on automatic scan clean-up algorithms.

7. Conclusion and Future Work

We described an improved discrete exponential map approximation of local tangent-spaces and new tangent-space graph Laplacian weights. Using this machinery, we constructed spectral methods to generate Dirichlet and conformal parameterizations of point-set surfaces. Our approach also improves existing mesh-based techniques, and leads to robust free-boundary parameterizations with uniform, authalic, mean-value, and mixed weight matrices.

It has recently been shown that while *heat-kernel* (Gaussian) Laplacians with a fixed Euclidean ε-ball radius converges to $\triangle_{\mathcal{S}}$ as sampling density increases, the cotangent Laplacian does not [BSW08]. Essentially, convergence occurs because the local integral takes an increasing number of samples into account, where the cotangent Laplacian is limited to a mesh one-ring, However, the heat-kernel weights do not minimize reconstruction error, and generate low-quality parameterizations. Our tangent-space graph Laplacian weights share some of the "geometry-preserving" properties of the cotangent weights, while also being able to take an increasing number of samples into account, and our results indicate that the tangent-space weights provide a reasonable approximation of $\triangle_{\mathcal{S}}$. This suggests that it may be possible to derive a formulation which is representative of the geometry and also convergent.

References

[BCMB08] BEN-CHEN M. GOTSMAN C., BUNIN G.: Conformal flattening by curvature prescription and metric scaling. *Comp. Graph. Forum 27*, 2 (2008). 1

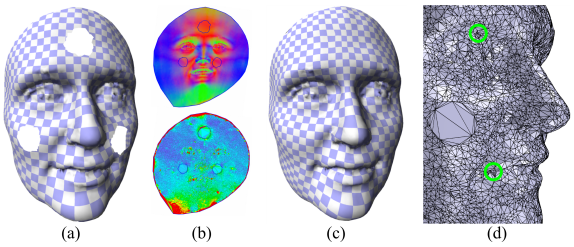


Figure 15: *Face mesh (10k pts) with holes. The result in (a) is computed with **SCP**, after flattening and meshing using **OCP** (b,c). Although highly irregular, the resulting mesh only has a few small self-intersections (d, green circles).*

[BN03] BELKIN M., NIYOGI P.: Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Comput. 15*, 6 (2003), 1373–1396. 2, 3, 5

[BSK06] BENDELS R., SCHNABEL R., KLEIN R.: Detecting holes in point set surfaces. *Journal of WSCG* (2006). 6

[BSW08] BELKIN M., SUN J., WANG Y.: Discrete laplace operator on meshed surfaces. In *Proc SOCG '08* (2008), pp. 278–287. 9

[BZK09] BOMMES D., ZIMMER H., KOBBELT L.: Mixed-integer quadrangulation. *ACM Trans. Graph. 28*, 3 (2009), *Article* 77. 8

[CLZW07] CHEN Z., LIU L., ZHANG Z., WANG G.: Surface parameterization via aligning optimal local flattening. In *Proc. SPM '07* (2007), pp. 291–296. 2

[DG03] DONOHO D., GRIMES C.: Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. *Proc. PNAS 100*, 10 (2003), 5591–5596. 2

[DMA02] DESBRUN M., MEYER M., ALLIEZ P.: Intrinsic parameterizations of surface meshes. *Comp. Graph. Forum 21* (2002), 209–218. 1, 2, 3, 8

[dST02] DE SILVA V., TENENBAUM J.: Unsupervised learning of curved manifolds. In *Nonlinear Estimation and Classification*. Springer-Verlag, 2002, pp. 453–466. 2

[FH02] FLOATER M., HORMANN K.: Parameterization of triangulations and unorganized points. In *Tutorials on Multiresolution in Geometric Modelling*. Springer, 2002, pp. 287–316. 1

[FH05] FLOATER M., HORMANN K.: Surface parameterization: a tutorial and survey. In *Advances in Multiresolution for Geometric Modelling*. Springer, 2005, pp. 157–186. 1

[Flo03] FLOATER M. S.: Mean value coordinates. *Comp. Aided Geom. Design 20* (2003), 19–27. 3, 8

[FR01] FLOATER M., REIMERS M.: Meshless parameterization and surface reconstruction. *Comp. Aided Geom. Design 18* (2001), 77–92. 1

[GP07] GROSS M., PFISTER H. (Eds.): *Point-Based Graphics*. Morgan Kauffman, 2007. 1

[JKG07] JIN M., KIM J., GU X.-D.: Discrete surface ricci flow: theory and applications. *Mathematics of Surfaces XII* (2007). 1

[KB04] KOBBELT L., BOTSCH M.: A survey of point-based techniques in computer graphics. *Comp. & Graph. 28*, 6 (2004), 801–814. 1

[KSS06] KHAREVYCH L., SPRINGBORN B., SCHRÖDER P.: Discrete conformal mappings via circle patterns. *ACM Trans. Graph. 25*, 2 (2006), 412–438. 1, 8

[LPRM02] LÉVY B., PETITJEAN S., RAY N., MAILLOT J.: Least squares conformal maps for automatic texture atlas generation. *ACM Trans. Graph. 21*, 3 (2002), 362–371. 1

[LYD*05] LIU Y.-S., YU P.-Q., DU M.-C., YONG J.-H., ZHANG H., PAUL J.-C.: Mesh parameterization for an open connected surface without partition. In *Proc. CAD-CG '05* (2005). 2

[MTAD08] MULLEN P., TONG Y., ALLIEZ P., DESBRUN M.: Spectral conformal parameterization. *Comput. Graph. Forum 27*, 5 (2008), 1487–1494. 1, 2, 3, 6, 7, 8

[PC05] PEYRÉ G., COHEN L.: Geodesic computations for fast and accurate surface remeshing and parameterization. *Progress in Nonlinear Differential Equations and Applications 63* (2005). 2

[PP93] PINKALL U., POLTHIER K.: Computing discrete minimal surfaces and their conjugates. *Exper. Math. 2*, 1 (1993), 15–36. 1, 2, 3, 5

[RS00] ROWEIS S., SAUL L.: Nonlinear dimensionality reduction by locally linear embedding. *Science 290*, 5500 (2000), 2323–2326. 2, 3, 4

[SGW06] SCHMIDT R., GRIMM C., WYVILL B.: Interactive decal compositing with discrete exponential maps. *ACM Trans. Graph. 25*, 3 (2006), 605–613. 2, 3, 4

[SJGT06] S. J. GORTLER C. G., THURSTON D.: Discrete one-forms on meshes and applications to 3d mesh parameterization. *Journal of CAGD 33*, 2 (2006), 83–112. 3

[SLMB05] SHEFFER A., LÉVY B., MOGILNITSKY M., BOGOMYAKOV A.: ABF++: Fast and robust angle based flattening. *ACM Trans. Graph.* (2005). 1

[SPR06] SHEFFER A., PRAUN E., ROSE K.: Mesh parameterization methods and their applications. *Found. Trends. Comput. Graph. Vis. 2*, 2 (2006), 105–171. 1

[SSP08] SPRINGBORN B., SCHRÖDER P., PINKALL U.: Conformal equivalence of triangle meshes. *ACM Trans. Graph. 27*, 3 (2008), Article 77. 1

[TdSL00] TENENBAUM J., DE SILVA V., LANGFORD J. C.: A global geometric framework for nonlinear dimensionality reduction. *Science 290*, 5500 (2000), 2319–2323. 2

[WS04] WEINBERGER K., SAUL L.: Unsupervised learning of image manifolds by semidefinite programming. In *Proc. CVPR '04* (2004), vol. 2, pp. 988–995. 2

[WSZS07] WEINBERGER K., SHA F., ZHU Q., SAUL L.: Graph laplacian methods for large-scale semidefinite programming, with an application to sensor localization. In *Advances in Neural Information Processing Systems 19*. MIT Press, 2007. 2

[ZG04] ZWICKER M., GOTSMAN C.: Meshing point clouds using spherical parameterization. In *Proc. PBG '04* (2004). 2

[ZKK02] ZIGELMAN G., KIMMEL R., KIRYATI N.: Texture mapping using surface flattening via multi-dimensional scaling. *IEEE Trans. Vis. Comp. Graph. 8*, 2 (2002), 198–207. 2

[ZPKG02] ZWICKER M., PAULY M., KNOLL O., GROSS M.: Pointshop 3d: an interactive system for point-based surface editing. In *Proc. SIGGRAPH '02* (2002), pp. 322–329. 2, 8

[ZvKD09] ZHANG H., VAN KAICK O., DYER R.: Spectral mesh processing. *Computer Graphics Forum* (2009), to appear. 2