AN ANISOTROPIC MESH PARAMETERIZATION SCHEME

Igor Guskov

University of Michigan, Ann Arbor
guskov@eecs.umich.edu

ABSTRACT

We introduce a simple anisotropic modification of the Floater's shape-preserving parameterization scheme. The original scheme is formulated as a discrete energy minimization and the modification is performed by introducing an additional stretching term. Results and example applications to anisotropic regular surface meshing are presented.

Keywords:structured surface mesh generation, anisotropy, remeshing, surface parameterization

1. INTRODUCTION

Surface meshes are widely used in manufacturing, medical and scientific applications. Acquired with shape acquisition techniques, these meshes are often resampled into more regular representations [1] [2] to become more amenable for further usage. Creation of such regular representations is called remeshing, and it often involves patchwise parameterization of original mesh data onto simple planar regions. Recently, there has been a bout of interest in surface mesh parameterization algorithms targeting surface texturing [3] [4], geometry approximation with semi-regular approximations [5] [6], as well as general mesh parameterization techniques [7][8].

In this paper we introduce a modification to a well-known shape-preserving parameterization scheme of Michael Floater [9]. We work in a setting useful to traditional remeshing algorithms that split the original surface mesh into topologically simple patches, and map each patch onto a simple planar region. We therefore restrict our attention to the case of a single mesh region mapped onto a square.

The goal of this paper is to introduce more control over the sampling of the remeshed model. In particular, our method creates rarefied sampling according to a given direction field. Since the parametric region (a unit square) is fixed, that will incur a denser sampling in other regions of the surface. We can therefore ad-

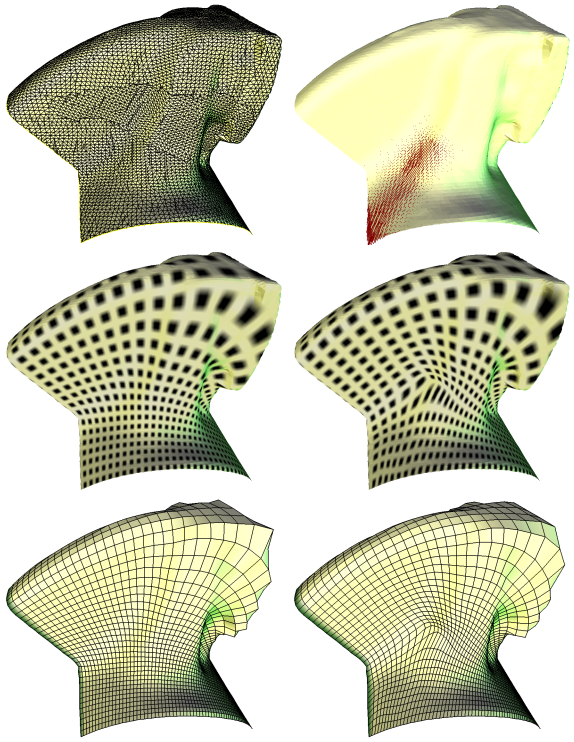

Figure 1: ***The Floater's original isotropic and our anisotropic parameterizations of a patch of the Rockerarm mesh. Top: original mesh and a direction field; middle: parameterizations; bottom: resulting regular remeshes.***

just surface sampling in a desired way. Figure 1 shows an example of a controlled stretching of the parameterization away from a corner of the Rockerarm mesh patch.

This research originated from the desire to have more control over parameterization without introduction of rigid constraints on the parameterization. This can be useful in applications where the regular mesh density needs to be adjusted: instead of specifying the regions where mesh needs to be more dense, we specify the regions where the regular sampling can be sparser in a certain direction.

A possible application of our scheme can produce better surface approximations by optimizing the direction field to sparsify regions with low curvature to pull the sampling of the reconstructed regular meshes towards the areas with higher curvature. An alternative approach to this problem was recently presented by [4].

In the following sections we formulate a variant of the Floater's scheme in terms of second difference minimization similar to [10]. This allows us to incorporate anisotropic stretching modification in a natural way by adding another term to the minimization functional.

2. MESH PARAMETERIZATION

Notation We consider a triangular manifold mesh $\mathcal{M} = (\mathcal{V}, \mathcal{T})$ with the vertex set $\mathcal{V}$ and the face set $\mathcal{T}$, and a "coordinate" function $\mathbf{x} : \mathcal{V} \to \mathbf{R}^3$. When using a local parameterization on a small neighborhood of the mesh, we shall reserve $\xi = (\xi^1, \xi^2)$ for such parameterization. A typical local parameterization of the "umbrella" of faces adjacent to a given vertex can be obtained by flattening such a neighborhood via a "polar map" as described in [12] [9].

Parameterization bijectively maps a mesh region onto a planar region. In remeshing applications a regularly sampled mesh is the goal, and it is therefore typical that the boundary of a mesh patch is mapped onto the boundary of a simple plane region (e.g. a square) in a fixed way. The parameterization scheme is then often specified by a per-inner-vertex relations: both linear [9] [1] and non-linear [7] approaches are employed in practice. When linear equations are used, the bijectivity of the parametric mapping can be ensured by introducing a convexity condition.

Formally, the goal of parameterization is to find a parametric function $U : \mathcal{V} \to \mathbf{R}^2$, such that it maps the boundary vertices of the mesh $\mathcal{M}$ onto the boundary of a simple planar region (we shall use a unit square in this paper), while the inner vertices are mapped inside the square with the condition that the corresponding piecewise linear map based on U and $\mathcal{T}$ is injective (there are no flipped triangles in the parametric region). See [13] for more details. Technicalities aside, a sufficient condition for having a bijection between the planar square and the original mesh is that the parameteric function $U = (U^1, U^2)$ satisfies a convex relation at every inner vertex of the mesh, so that for every inner vertex $\bar{v}$ and every vertex v in its one-ring $\omega(\bar{v})$ there exist real weights $\alpha_{\bar{v}v}$ such that

$$\alpha_{\bar{v}v} > 0 \quad \text{(positivity)}$$
$$\sum_{v \in \omega(\bar{v})} \alpha_{\bar{v}v} = 1 \quad \text{(affineness)}$$

and

$$\sum_{v \in \omega(\bar{v})} \alpha_{\bar{v}v} U(v) = U(\bar{v}).$$

Such parameterization is called a *convex combination map* [13].

Hence, one approach to parameterization will be to specify a collection of weights satisfying the above condition, and then find the parameterization inside the mesh region by solving the resulting linear system. As we shall see below, the affiness condition is easy to ensure by restricting ourselves to a certain class of schemes, while the positivity condition is harder to achieve.

2.1 Second differences

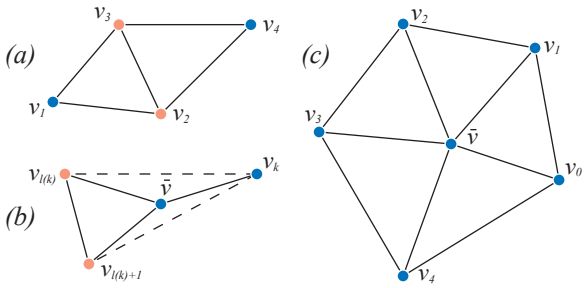

Figure 2: *(a) A typical $\Delta^{[2]}$-stencil from [10]; (b) a typical $\Delta^{[2]}$-stencil used in the Floater's scheme; (c) one-ring indexing for Floater's scheme.*

Consider a real function $f : \mathcal{V} \to \mathbf{R}$ (think of $f = U^1$ or $f = U^2$). Given an assignment of the local neighborhood parameterizations we treat f as samples of a function from $\mathbf{R}^2$ to $\mathbf{R}$. More precisely, let Ω be a neighborhood submesh where the local parameterization ξ is established. The parameterization ξ is thus defined on the vertex set $\mathcal{V}_\Omega$ of Ω (e.g. $\mathcal{V}_\Omega = \omega(\bar{v}) \cup \bar{v}$ for some vertex $\bar{v}$).

Given a triple of vertices $v_1, v_2, v_3 \in \mathcal{V}_\Omega$ we define the first divided difference $D^{[1]}_{\{v_1,v_2,v_3\}} f$ of f on the $D^{[1]}$-*stencil* $\{v_1, v_2, v_3\}$ as the slope of the linear interpolant passing through these three sample values; thus

$$D^{[1]}_{\{v_1,v_2,v_3\}} f = V^{-1} \Delta f,$$

where $\Delta f = [f_2 - f_1, f_3 - f_1]^T$ and the matrix V is given as

$$V = \begin{bmatrix} \xi_2^1 - \xi_1^1 & \xi_2^2 - \xi_1^2 \\ \xi_3^1 - \xi_1^1 & \xi_3^2 - \xi_1^2 \end{bmatrix},$$

where $\xi_k^i := \xi^i(v_k)$.

Given two $D^{[1]}$-stencils that share a pair of vertices it is easy to see that the difference between the corresponding first divided differences is always orthogonal to the line connecting the two common points. Thus, a scalar quantity measuring closeness of first derivatives of a sampled function can be introduced as a projection of the above difference onto the corresponding normal direction. More precisely, let $\{v_1, v_2, v_3\}$ and $\{v_2, v_3, v_4\}$ be the $D^{[1]}$-stencils under consideration. We define the unit vector orthogonal to the edge $(\xi(v_2), \xi(v_3))$ as

$$\mathbf{n}_{23} = [(\xi_3^2 - \xi_2^2)/\left\|\xi_3 - \xi_2\right\|, (\xi_2^1 - \xi_3^1)/\left\|\xi_3 - \xi_2\right\|].$$

The *second difference* $\Delta^{[2]}_{(v_1,v_2,v_3,v_4)} f$ is then defined via

$$\Delta^{[2]}_{(v_1,v_2,v_3,v_4)} f = \mathbf{n}_{23} \cdot (D^{[1]}_{\{v_1,v_2,v_3\}} f - D^{[1]}_{\{v_2,v_3,v_4\}} f) = c_1 f(v_1) + c_2 f(v_2) + c_3 f(v_3) + c_4 f(v_4),$$

where

$$c_1 = \frac{\|\xi(v_2) - \xi(v_3)\|}{S_{123}}, \quad c_2 = -\frac{\|\xi(v_2) - \xi(v_3)\| \, S_{314}}{S_{123} \, S_{432}},$$
$$c_3 = -\frac{\|\xi(v_2) - \xi(v_3)\| \, S_{241}}{S_{123} \, S_{432}}, \quad c_4 = \frac{\|\xi(v_2) - \xi(v_3)\|}{S_{432}},$$

and the signed areas S_{ijk} are given as

$$S_{ijk} := \det \begin{bmatrix} 1 & \xi^1(v_i) & \xi^2(v_i) \\ 1 & \xi^1(v_j) & \xi^2(v_j) \\ 1 & \xi^1(v_k) & \xi^2(v_k) \end{bmatrix}.$$

Put simply the second difference characterizes the change of slope across an edge shared by two triangles.

In [10] second differences where used in the construction of multiresolution subdivision filters on irregular mesh hierarchies, and the $D^{[1]}$-stencils were coming from triangles of the mesh. In the following section we show how the Floater's shape-preserving scheme can be constructed using second differences.

2.2 Floater's parameterization scheme

We follow Floater [9] and consider one ring neighborhood of a particular inner vertex of a triangular mesh. Let $\bar{v}$ be the center vertex of the one ring, and $\omega(\bar{v}) = \{v_0, \ldots, v_{n-1}\}$ be the set of its neighbors indexed consistently in a counter-clockwise order (see Figure 2). Floater considers a vertex $v_k \in \omega(\bar{v})$ and finds an index $l = l(k)$ such that the ray $\overrightarrow{\xi(v_k)\xi(\bar{v})}$ intersects the segment $conv\{\xi(v_l), \xi(v_{l+1})\}$ (all indices in one ring are treated modulo vertex valence n). This ensures that $\xi(\bar{v})$ lies inside the triangle $conv\{\xi(v_l), \xi(v_{l+1}), \xi(v_k)\}$. (See Figure 2(b)). At this point, we look at the second difference

$$g_{[2]}(\bar{v}, v_k) := \Delta^{[2]}_{(\bar{v}, v_{l(k)}, v_{l(k)+1}, v_k)} f = c_{\bar{v}}^{\bar{v},v_k} f(\bar{v}) + c_{v_k}^{\bar{v},v_k} f(v_k) + c_{v_l}^{\bar{v},v_k} f(v_l) + c_{v_{l+1}}^{\bar{v},v_k} f(v_{l+1}),$$

and notice that three coefficients $c_{v_k}^{\bar{v},v_k}, c_{v_l}^{\bar{v},v_k}, c_{v_{l+1}}^{\bar{v},v_k}$ have the same sign while $c_{\bar{v}}^{\bar{v},v_k}$ is of the opposite sign. With a bit more precision we claim the following is true:

$$c_{v_k}^{\bar{v},v_k} c_{\bar{v}}^{\bar{v},v_k} < 0, \quad c_{v_l}^{\bar{v},v_k} c_{\bar{v}}^{\bar{v},v_k} \le 0, \quad c_{v_{l+1}}^{\bar{v},v_k} c_{\bar{v}}^{\bar{v},v_k} \le 0. \quad (1)$$

Note: One of the $D^{[1]}$ stencils participating in $g_{[2]}$ is no longer aligned with a triangle from $\mathcal{T}$.

In order to obtain a linear predictor $P_{\bar{v}}$ for the function value at the center vertex $\bar{v}$ given the neighboring values we minimize the following functional which is quadratic in $f(\bar{v})$ (we label it J^{iso} for being isotropic):

$$J^{iso}_{\bar{v}}(f) = \sum_{k=0}^{n-1} (g_{[2]}(\bar{v}, v_k))^2.$$

The desired value of the function at the center vertex is then given as

$$P_{\bar{v}} f = argmin_{f(\bar{v})} J^{iso}_{\bar{v}}(f).$$

One can easily check that the corresponding parameterization scheme is identical to the scheme of Floater [9]. Indeed, for the optimal value of $f(\bar{v})$ we have

$$\sum_k c_{\bar{v}}^{\bar{v},v_k} (c_{\bar{v}}^{\bar{v},v_k} f(\bar{v}) + c_{v_k}^{\bar{v},v_k} f(v_k) + c_{v_{l(k)}}^{\bar{v},v_k} f(v_{l(k)}) + c_{v_{l(k)+1}}^{\bar{v},v_k} f(v_{l(k)+1}) = 0,$$

Using the fact that the coefficients of the second difference operator always sum up to zero, we obtain the following linear relation

$$(\sum_s A_s^{iso}) f(\bar{v}) = \sum_s A_s^{iso} f(v_s),$$

where the coefficients A_s^{iso} are given by

$$A_s^{iso} = -c_{\bar{v}}^{\bar{v},v_s} c_{v_s}^{\bar{v},v_s} - \sum_{k:s \in \{l(k), l(k)+1\}} c_{\bar{v}}^{\bar{v},v_k} c_{v_s}^{\bar{v},v_k}.$$

Note that A_s^{iso} are guaranteed to be positive because of (1).

Thus, introducing $\alpha_s^{iso} = A_s^{iso} / \sum_s A_s^{iso}$ we get the center vertex function value as a convex combination of surrounding values:

$$f(\bar{v}) = \sum_s \alpha_s^{iso} f(v_s).$$

2.3 General formulation

The Floater parameterization scheme described above considers a collection of local functionals that characterize certain function properties (such as smoothness) and minimize these quantities in the least squares sense. In the following section we consider a different (enlarged) set of such local functionals, and it will pay off to derive the result of such minimization problem in the general case. For a similar discussion, see [11].

Let $M(\bar{v})$ be a collection of stencils each of which includes a fixed vertex $\bar{v}$. Each stencil τ from $M(\bar{v})$ is provided with a set of coefficients defining a linear functional L_τ via $L_\tau f := \sum_{v\in\tau} \lambda_{\tau,v} f(v)$. We assume that all L_τ's used in our constructions annihilate constants, so that

$$\sum_{v\in\tau} \lambda_{\tau,v} = 0. \tag{2}$$

We also assume that these stencils cover the one-ring of the vertex $\bar{v}$, that is

$$\bigcup_{\tau\in M(\bar{v})} \tau = \omega(\bar{v}) \cup \{\bar{v}\}.$$

In order to find the function value at the center vertex, we form $J_{\bar{v}}(f) := \sum_{\tau\in M(\bar{v})} (L_\tau f)^2$ and minimize it with respect to the function value at the "center" vertex $f(\bar{v})$.

Differentiating $J_{\bar{v}}$ with respect to $f(\bar{v})$, it is easy to see that the resulting scheme should have the following form:

$$\sum_{\tau\in M(\bar{v})} \left(\lambda_{\tau,\bar{v}} f(\bar{v}) + \sum_{v\in\tau\setminus\{\bar{v}\}} \lambda_{\tau,v} f(v) \right) \lambda_{\tau,\bar{v}} = 0.$$

Rearranging the terms and using (2) we obtain:

$$\Big(\sum_{v\in\omega(\bar{v})} A_v \Big) f(\bar{v}) = \sum_{v\in\omega(\bar{v})} A_v f(v),$$

where

$$A_v := - \sum_{\tau\in M(\bar{v}):\tau\ni v} \lambda_{\tau,v}\lambda_{\tau,\bar{v}}. \tag{3}$$

It is immediately clear that the affineness condition always holds for the schemes constructed in this way. On the other hand a sufficient condition for the scheme to have positive coefficients is that every contributing stencil has its coefficients for non-center vertices to be of the opposite sign to its coefficient for the center vertex. This condition holds for the Floater's scheme construction in the previous section.

Example A simple example of a parameterization scheme can be obtained by considering the set of functionals

$$\{\Delta_i f := f(v_i) - f(\bar{v}) : i = 0, \ldots, n-1\}.$$

Then the corresponding stencils are $\tau_i := \{\bar{v}, v_i\}$ and the coefficients λ are given as

$$\lambda_{\tau_i,\bar{v}} = -1, \quad \lambda_{\tau_i,v_i} = 1.$$

Hence, $A_v = 1$ and the resulting parameterization scheme is a simple regular umbrella:

$$nf(\bar{v}) = \sum_{i=0}^{n-1} f(v_i).$$

3. ANISOTROPIC MODIFICATION

In this section we introduce a modification to the original Floater's scheme that can produce anisotropic parameterization stretched in a direction given by a direction field on the surface. We shall use a direction field specified as a vector on each face of the mesh. The direction fields can be specified by user or created automatically. In our implementation user specifies a desired rough direction field, which is then smoothed using a procedure similar to the approach presented in [14].

3.1 A simple scheme

The direction field is given by a vector represented in the local coordinate system of each triangle of the mesh. It turns out that the fact that the vector field is directed will not matter for the derivation of the anisotropic scheme. Formally, after the local neighborhood parameterization ξ is fixed, we have an assignment of direction vectors to triangles $h : \mathcal{T} \to \mathbf{R}$ that samples a vector field $H = H^i \partial/\partial\xi^i$. The anisotropic scheme will stretch both parametric functions $U^j, j = 1, 2$ in the direction of H. To achieve that effect we add an extra term representing $(HU^j)^2$ into the minimization of $J(U^j)$. As the result the derivative in the direction of H will get smaller, introducing a stretch along the given direction field. Note that the "negated" vector field $-H$ will result in the same stretch, and thus the directionality of H does not matter at the parameterization stage (it does matter in the vector smoothing step that produces H, see [14]).

The discrete implementation of the above approach is straightforward: we replace the partial derivatives by the first divided difference operator and add the sum of squares of the resulting quantities to the isotropic functional. This yields an anisotropic functional J^{anis}

$$J^{anis}_{\bar{v},\beta}(f(\bar{v})) = \sum_{k=0}^{n-1} \left([g_{[2]}(\bar{v}, v_k)]^2 + \beta [f_{[1],h}(\bar{v}, v_k, v_{k+1})]^2 \right),$$

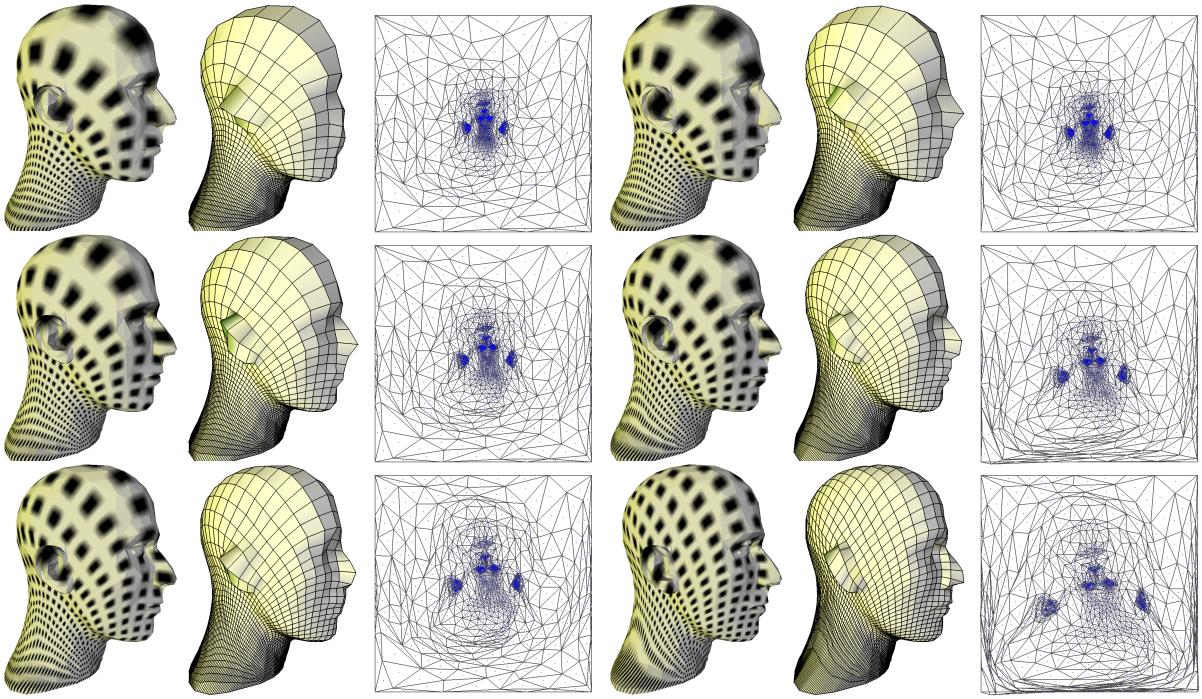

Figure 3: ***Clamped anisotropic (left) and non-clamped anisotropic (right) schemes for the Mannequin model. Values of*** β ***are*** 100***(top),*** 400***(middle),*** 1600***(bottom). It is easy to see the non-convexity for the non-clamped scheme (extreme parametric distortions on the back of the head and hence denser mesh in the face region). The positivity correction of the clamped scheme guarantees the bijective mapping.***

where the first part of the sum is copied from the isotropic case and the quantities in the second term are defined for a general vertex triple $t = \{v_0, v_1, v_2\}$ and an associated direction vector h_t via

$$f_{[1],h}(v_0, v_1, v_2) = h(\{v_0, v_1, v_2\}) \cdot D^{[1]}_{\{v_0,v_1,v_2\}} f = d_0 f(v_0) + d_1 f(v_1) + d_2 f(v_2),$$

where

$$d_0 = \frac{1}{S_{012}} \left[(\xi^2(v_1) - \xi^2(v_2)) h_t^1 + (\xi^1(v_2) - \xi^1(v_1)) h_t^2 \right],$$
$$d_1 = \frac{1}{S_{012}} \left[(\xi^2(v_2) - \xi^2(v_0)) h_t^1 + (\xi^1(v_0) - \xi^1(v_2)) h_t^2 \right],$$
$$d_2 = \frac{1}{S_{012}} \left[(\xi^2(v_0) - \xi^2(v_1)) h_t^1 + (\xi^1(v_1) - \xi^1(v_0)) h_t^2 \right].$$

We can then repeat the derivation of the previous section to obtain

$$(\sum_s A_s^{anis}) f(\bar{v}) = \sum_s A_s^{anis} f(v_s),$$

with A_s^{anis} derived from (3).

So far we took no special care to insure the positivity of the coefficents A_s^{anis}, hence there is no guarantee that the resulting linear system is well-defined. It is clear however that given a bounded direction field, $J^{anis}_{\bar{v},\beta}$ will produce a convex combination map scheme for small values of β. As we explain in the next section, clamping the value of β to the maximum allowed by positivity condition, turns out to produce a very practical anisotropic parameterization scheme.

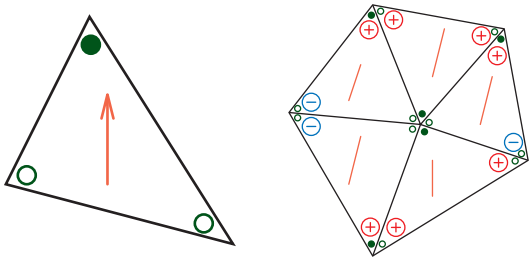

Figure 4: ***Left: the signs of the coefficients for*** $f_{[1],h}$ ***operator. Right: The signs of the contributions to*** $A_s^{stretch}$ ***coefficients for the given distribution of direction vectors.***

3.2 Positivity

In this section we discuss possible approaches to making the scheme monotone. First, we look at the stencil for directed derivatives. The main issue is the signs of the coefficients. Let $\tau = \{\bar{v}, v_1, v_2\}$ be a stencil of three vertices. Then it will contribute monotonely to

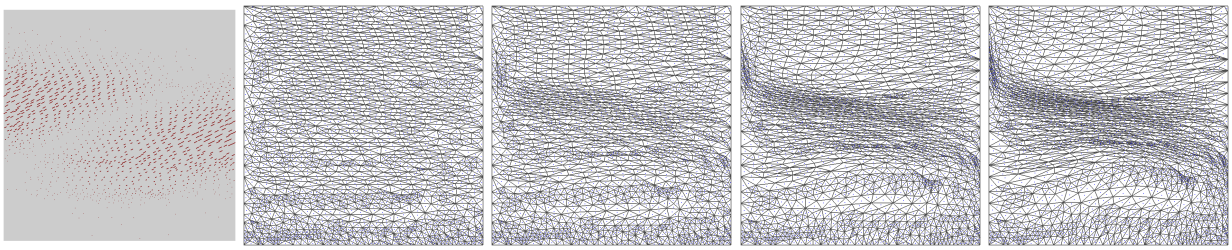

Figure 5: ***Square anisotropic parameterization for*** $\beta = 0, 800, 6400, 25600$***. The direction field is on the left, followed by the meshes mapped onto parametric plane.***

the scheme centered at $\bar{v}$ if the direction h splits the angle $\xi(v_1)\widehat{\xi(\bar{v})}\xi(v_2)$. Formally, a vector $h = (h^1, h^2)$ splits an angle $\widehat{\xi\eta\zeta}$ if $h^\perp \cdot (\xi - \eta)$ and $h^\perp \cdot (\zeta - \eta)$ are of opposite signs (we use $h^\perp = (h^2, -h^1)$). Note that within a triangle, only one of the angles is split by a given direction vector (this angle is denoted with a filled circle in Figure 4). The coefficient of $f_{[1],h}$ for the split corner is opposite to the sign of the other two coefficients (this can be seen visually by varying the function values one-by-one, since $f_{[1],h}$ gives a slope of the function in the given direction).

One conservative way to ensure the positivity of the scheme is to make sure that a triangle's anisotropic term only contributes to $J^{anis}_{\bar{v},\beta}$ for the vertex $\bar{v}$ whose corner is split by the direction given on that triangle. This is easy to implement, however our experiments show that the resulting scheme does not produce sufficient stretching even for large values of β.

We therefore adapt a less conservative and somewhat simpler approach that is mentioned in the previous section. Namely, for every inner vertex $\bar{v}$ we clamp the value of β to be less than the precomputed value β_{max} that ensures the positivity of the scheme at this particular vertex $\bar{v}$. We can find such $\beta_{max}(\bar{v})$ by noting that the coefficients A^{anis}_s are combinations of the isotropic Floater's coefficients and the "stretching" coefficients (we can do so by splitting the stencils of (3) into two groups). Hence, we obtain

$$A^{anis}_s = A^{iso}_s + \beta A^{stretch}_s,$$

where some of $A^{stretch}_s$ can be negative. We then find

$$\beta_{max}(\bar{v}) = \min_s \frac{A^{iso}_s}{-min(0, A^{stretch}_s)},$$

(note that we get ∞ value if all the coefficients $A^{stretch}_s$ are positive with a meaning that there is no restriction on $\beta(\bar{v})$.

Then the new "clamped anisotropic" scheme is obtained from

$$J^{ca}_{\bar{v},\beta}(f) := J^{iso}_{\bar{v}}(f) + \min(\beta, \beta_{max}(\bar{v}))J^{stretch}_{\bar{v}}(f),$$

where $J^{stretch}_{\bar{v}}(f) = \sum_{k=0}^{n-1}(f_{[1],h}(\bar{v}, v_k, v_{k+1}))^2$.

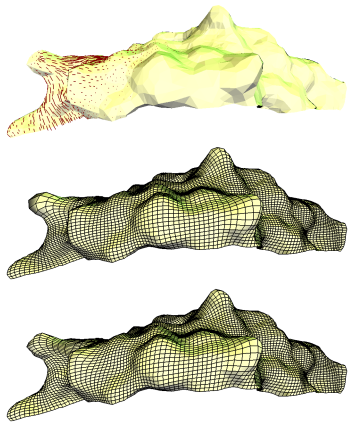

Figure 6: ***Focusing the sampling onto a high curvature geometric feature for a Molecule mesh patch. Top: direction field on the original surface. Middle: isotropic remesh. Bottom: anisotropic remesh using the given direction field for*** $\beta = 400$***.***

3.3 Results

We demonstrate the performance of the schemes described above on a number of examples. The parameterizations produced with our scheme are visualized in a number of ways: as the mapping of a regular texture using produced parametric functions, as the resampling of the original surface, and as the picture of the original mesh in the parametric (U^1, U^2)-plane.

Figure 3 compares the performance of the clamped scheme that guarantees the convexity condition, with that of the non-clamped scheme of Section 3.1. It is clear that for the same value of parameter β the clamped scheme will have less stretching than the non-clamped one. Thus similar parameterization shift (and change in regular sampling frequency) is achieved at different values of β for the considered two schemes. It is also clear that the non-clamped scheme results in some non-injective parameterizations (note the triangles mapped outside of the unit square in the right column bottom row). Note that the effects of the clamped scheme is milder, which results in a sparser mesh on the mannequin face (thus its features are less resolved). For the non-clamped scheme high values of

β result in extreme rarefication of the back portion of head, thus a denser mesh with better resolved features appears in the face region.

Figure 5 illustrates the effect of the anisotropic stretching in the parametric plane. The parametric function becomes "flatter" in the given directions, which results in bringing the points closer in the parametric domain. Thus, we see a close packing of squeezed triangles in a certain region of the triangulation. The role of the convexity condition is therefore to keep the triangles from flipping and riding on top of other triangles. Note that the case $\beta = 0$ is the original Floater scheme which preserves planar triangulations, so that the leftmost mesh is the original one.

Figure 6 uses the direction field to focus the sampling onto a high curvature feature of the Molecule model. The direction field vectors point away from the feature and are null on the feature itself. This results in allowing more samples to be placed onto the feature resulting in a better approximation.

Figure 7 shows a simple way to introduce a wiggle pattern for the parameterization lines. Two stretching regions push the samples of the remeshed model away from them.

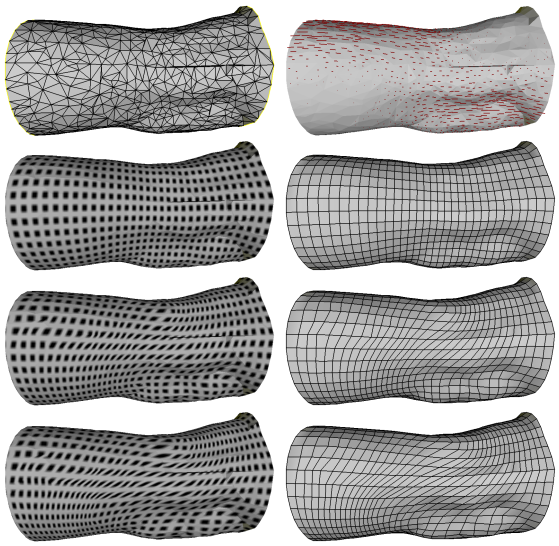

Figure 7: ***Clamped anisotropic scheme for the Knee model: the first row shows the original mesh and the direction field, the second row shows the parameterization and regular remesh for*** $\beta = 0$***, the third row for*** $\beta = 1000$***, and the fourth row for*** $\beta = 10000$***.***

Performance note The introduced schemes produce a linear system of equations that is solved with a biconjugate gradient method. Since the mesh sizes are small (tens of thousands of vertices maximum) all the computations take a matter of seconds.

4. CONCLUSIONS

We have introduced a simple anisotropic modification of the Floater's shape-preserving parameterization scheme that allows flattening of parameteric mapping along a given direction field. The future work includes development of algorithms for constructing direction fields that allow better approximation of geometric features as well as construction of unconditionally positive schemes. Another interesting question is whether one can improve the scheme performance via selective local mesh refinement. Currently there are no estimates on the shift in the parameterization in relation to the changes in the anisotropic parameter β. Finding such relation is important for applying our scheme to automated parameterization adjustment. Extension of our approach to 3D parameterization is also a possibility.

In order to be applicable to meshes of arbitrary topology, our method needs to be combined with a global layout generation procedure such as described in [5]: the parameterization of simple regions is an inherent part of such a procedure, and a globally specified direction field will affect the distribution of patch boundaries in a natural way.

Acknowledgements This work was supported in part by NSF(CCR-0133554). Models are courtesy Cyberware, University of Washington, and the Scripps Institute.

References

[1] Hoppe H., DeRose T., Duchamp T., Halstead M., Jin H., McDonald J., Schweitzer J., Stuetzle W. "Piecewise Smooth Surface Reconstruction." *Proceedings of SIGGRAPH*, pp. 295–302, 1994

[2] Krishnamurthy V., Levoy M. "Fitting Smooth Surfaces to Dense Polygon Meshes." *Proceedings of SIGGRAPH*, pp. 313–324, 1996

[3] Praun E., Finkelstein A., Hoppe H. "Lapped textures." *Proceedings of SIGGRAPH*, pp. 465–470, 2000

[4] Sander P.V., Snyder J., Gortler S.J., Hoppe H. "Texture Mapping Progressive Meshes." *Proceedings of SIGGRAPH*, pp. 409–416, 2001

[5] Guskov I., Vidimče K., Sweldens W., Schröder P. "Normal Meshes." *Proceedings of SIGGRAPH*, pp. 95–102, 2000

[6] Alliez P., Meyer M., Desbrun M. "Interactive Geometry Remeshing." *Proceedings of SIGGRAPH*, 2002

[7] A.Sheffer, de Sturler E. "Surface Parameterization for Meshing by Triangulation Flattening." *Proc. 9th International Meshing Roundtable*, pp. 161–172. 2000

[8] Lévy B., Mallet J. "Non-Distorted Texture Mapping for Sheared Triangulated Meshes." *Proceedings of SIGGRAPH*, pp. 343–352, 1998

[9] Floater M.S. "Parameterization and Smooth Approximation of Surface Triangulations." *Computer Aided Geometric Design*, vol. 14, 231–250, 1997

[10] Guskov I., Sweldens W., Schröder P. "Multiresolution Signal Processing for Meshes." *Proceedings of SIGGRAPH*, pp. 325–334, 1999

[11] Lévy B. "Constrained Texture Mapping for Polygonal Meshes." *Proceedings of SIGGRAPH*, pp. 417–424, 2001

[12] Lee A.W.F., Sweldens W., Schröder P., Cowsar L., Dobkin D. "MAPS: Multiresolution Adaptive Parameterization of Surfaces." *Proceedings of SIGGRAPH*, pp. 95–104, 1998

[13] Floater M.S. "Convex combination maps." *Algorithms for Approximation IV*. Huddersfield, 2001

[14] Hertzmann A., Zorin D. "Illustrating smooth surfaces." *Proceedings of SIGGRAPH*, pp. 517–526, 2000