



# Shear-Reduced Seamless Parametrization

Zohar Levi

*Victoria University of Wellington, New Zealand*

---

## Abstract

Shear distortion has been a popular formal and informal measure, whether used quantitatively or visually, in assessing the quality of mappings or the shape of elements in a quad mesh. Nevertheless, well-known energies such as conformal- and isometric-based energies do not directly target lowering shear distortion but only bound it. Consequently, the shear distortion can be unnecessarily high.

We introduce a new shear energy and offer an efficient way to incorporate it in the latest state-of-the-art optimization framework. The resultant mapping has substantially lower shear distortion, and the cost is a reasonably low addition of conformal or isometric distortion. The energy is minimized efficiently, and the run time of an iteration is of the same order of optimizing other popular energies. We also introduce a new scale-invariant, second-order smoothness energy that when combined with the shear energy, leads to smooth anisotropic mappings with low shear distortion.

We demonstrate these energies and compare with the state of the art in the application of seamless parametrization, where the quality of mapping a checkerboard pattern is paramount since it directly affects the quality of an extracted quad mesh.

© 2023 Published by Elsevier Ltd.

---

## 1. Introduction

A popular approach to quad meshing is via seamless parametrization using a cross field [1, 2, 3]. A seamless parametrization induces a flat metric over the surface, where the Gaussian curvature vanishes all over the surface except for a few points, termed cone singularities. The steps of the pipeline:

- A smooth cross field is generated over the mesh. In addition to a cross per face, a field also defines a matching per edge (or a period jump) that uniquely determines the cones.
- Seamless parametrization is generated from the field. The surface is cut into a disk, where the seam passes through all cones, and each seam edge is mapped into two twin edges in the domain, subject to seamlessness constraints: Vectors (the difference between end vertices) of two mapped twin edges (in the plane) differ by a  $90^\circ$ -multiple rotation. The crosses are used as guiding target frames when optimizing the mapping.
- Cone positions and translations of twin edges are rounded [4].
- A quad mesh (similar to a checkerboard texture pattern) with similar singularities is extracted.

The quality of the generated quad mesh is determined by the quality of the mapping, where high-quality mappings exhibit low isometric or conformal distortion. Such mappings are generated by optimizing corresponding energies. While these energies can also serve as quality measures, it is common to use measures that are specific to the shape of a quadrangle [5].

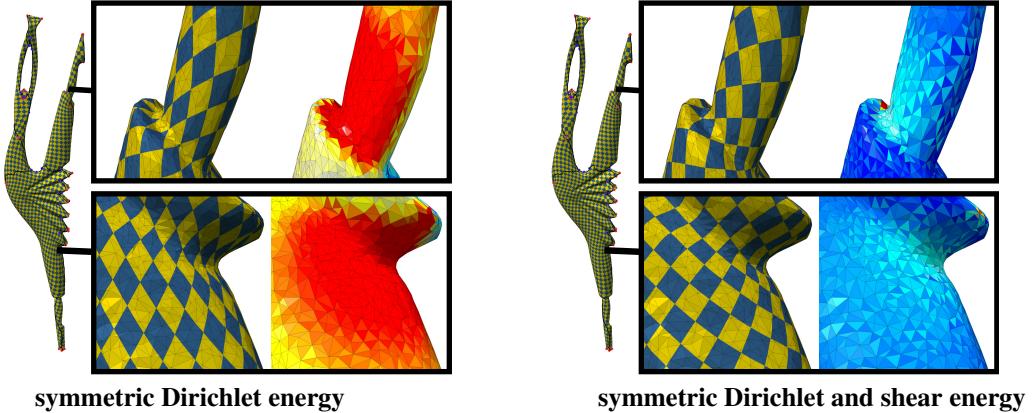


Figure 1. Minimizing the symmetric Dirichlet energy (*left*) exhibits quads with a substantial amount of shearing. Our added shear energy (*right*) results in quads that are more rectangular for the price of a reasonably low increase in isometric distortion. For statistics, see Figure 6.

One of these measures is the amount of shearing (or skewness). A few specific applications implicitly encourage shearing by preferring diagonal alignment of quads instead of orthogonal alignment. For example, creating the “fishnet stockings” in Figure 8 in [6] or the Chebyshev nets in [7]. However, the typical application prefers orthogonal alignment of quads with minimal shearing.

For example, the preference for the edge flow in a quad mesh is usually based on shape-awareness, capturing and representing the local shape adequately. The intention is usually a flow that follows orthogonal directions. Such an edge flow can be directed by the user [8, 9] or follow principal curvature directions [1, 10]. Principal curvature directions are orthogonal and support the ideal of rectangular faces with vanishing amount of shear. In the inset, the mapping of the cylinder on the right is based on the curvature. The mapping on the left is a rotation of the mapping on the right by 20°. Shearing is introduced, and the quads are no longer rectangular and do not follow principal curvature directions.



Another point that motivates low shear distortion is that orthogonality is inherent in cross field-based quad meshing methods. The cross represents orthogonal directions, and energies measure the difference between the Jacobian of a seamless mapping and a cross. The intention is, then, that the mapping would align with orthogonal directions that minimize shear distortion.

Another aspect that favors low shear distortion is maximizing the minimal angle. For a triangle mesh, Delaunay triangulation, which maximizes the minimal triangle angle, is desired in most cases. More generally, the ideal is equilateral triangles that maximize the minimal angle. Similarly, for a quad mesh, right angles maximize the minimal angle in a quadrangle.

The quality of an element is not only an aesthetic choice. It also affects the interpolation accuracy of a piece-wise linear mesh and the speed, accuracy, and stability of a finite element method [11]. A recent study [12] ranked the quality measures in [5] for hexahedral meshes, based on their correlations with the accuracy and stability metrics for simulations that solve a number of elliptic PDE problems. One of the conclusions was that skewness ranked highest among the metrics.

Different distortion measures were suggested in the past to measure the amount of shearing (or skewness) explicitly [13, 14, 15] or implicitly. For example, one common distortion measure is the scaled Jacobian; see Section 2 for related work and Section 4.1 for a discussion and comparison with our shear distortion measure. Minimizing shear-based energies, however, is less common. More specifically generating a seamless mapping (following a guiding cross field) while explicitly targeting and optimizing the amount of shearing has not been done before.

This oversight could be due to an attribute that is associated with conformal mapping, namely angle preservation (which distinguishes it from, e.g., isometric mapping, which is also area preserving). Angle preservation means that right angles of squares (e.g. in a checkerboard pattern) in the domain are preserved when pulled back to the surface via conformal mapping. And indeed, a conformal mapping has zero shear. Nevertheless, a conformal energy, which is minimized when optimizing a quasiconformal mapping, does not target shearing but only bounds it.

In conformal mappings, the problem starts with discretization. In the continuous setting, conformal mappings

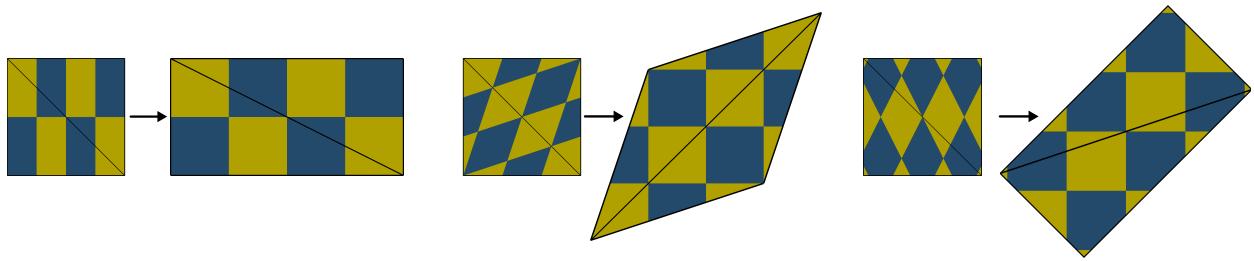


Figure 2. Problem motivation. (*Left*) mapping a square into a rectangle (ratio 1:2) by applying anisotropic scaling ( $0^\circ$  shear distortion). (*Middle*) mapping a square into a parallelogram by applying a  $45^\circ$  rotation before and after the scaling ( $37^\circ$  shear distortion). (*Right*) mapping a square into a rectangle rotated by  $45^\circ$  ( $37^\circ$  shear distortion). A checkerboard pattern is pulled back to the surface via the parametrization. All three mappings contain the same amount of isometric and conformal distortion, illustrating that these measures are oblivious and therefore susceptible to shear distortion (within the bound they are imposing).

always exist. There are several ways to discretize conformal maps, each of which fails to capture the behavior of smooth conformal maps in some essential way [16]. As a consequence, conformal distortion is in general inevitable in meshes, and a mapping can only be quasiconformal (excluding the case of a developable surface with vanishing Gaussian curvature). Discretization, though, is not the only issue. A perfectly conformal parametrization with bounded local area distortion—which can be detrimental in many applications—does not exist even in the continuous setting. Quasiconformal mappings with a bounded amount of distortion are more practical [17], and they may contain abundant shear distortion. Similarly for isometric mappings.

To optimize for a quasi-conformal or an as-isometric-as-possible mapping, a variational approach minimizes a related conformal or an isometric energy. These energies do not promote shearing reduction but only bound how much it can increase. Within the imposed bound, the energies are oblivious and susceptible to shear distortion, which can range freely from zero to the imposed bound. When conformality or isometry are reached, the bound on shear distortion is zero. As conformal or isometry distortion increases, the bound on shear distortion rises. Then, shear distortion can range from zero to as high as the bound if care is not taken. For illustration of this point, see Figures 2 and 3, and details are given in Section 4.2. Allowing anisotropy does not alleviate the problem.

We offer a new shear energy, which can be incorporated alongside isometric and conformal energies. Our shear energy is minimized efficiently in the same run-time order as optimizing the other energies. The resultant mapping yields a reasonably low increase in the amount of isometric and conformal distortions for a significant decrease in shear distortion.

We also suggest a second-order smoothness energy as an alternative regularization, which can be combined with our shear energy to create smooth anisotropic mappings with low shear distortion.

We provide analytical expressions for the Jacobian and the Hessian matrices of our shear and smoothness energies, along with code that verifies them.

## 2. Related Work

The problem of seamless parametrization and the application for quad meshing is discussed in [18]. Herein, we review work related to shear distortion.

The shear measure appears in various forms when determining the quality of a finite element [5, 14, 15]. In continuum mechanics, the engineering shear strain is defined as the change in angle between incident edges of an infinitesimal, rectangular material element after elastic deformation.

Marinov et al. [19] propose a technique for quad-dominant remeshing. After the mesh is segmented into patches, a network of candidate curves is computed, whose intersection creates quad elements. A subset of curves is selected from the candidate set using combinatorial optimization, which maximizes the quality measure of the elements in a configuration. One of the factors in the energy is orthogonality. Specifically, it is the sum of deviations of the four angles in a quad from  $90^\circ$ .

The work in [20, 21, 22] addresses the planar case of quad meshing. The motivation for quad meshing is that while triangulation is more common, quadrilateral elements are preferred in finite element analysis due to their superior

performance (better approximation when solving PDEs). It is important, however, to guarantee the quality of the elements, e.g., bounding the angles. Given a planar curve, these methods produce a quad meshing of the interior while striving to bound the angles of the final mesh within  $[45^\circ, 135^\circ]$ .

The scaled Jacobian has been a popular measure for the quality of quad meshes. For example, it was used in [23, 5, 24, 25, 26], or more recently in the supplemental gallery in [27]. This measure is directly related to shear distortion; see Section 4.1.

Daniels et al. [28] perform quad mesh simplification. Results are evaluated via a comparison of angle distribution and scaled Jacobians measured on the original and simplified meshes. Their ideal is quad angles near  $90^\circ$  and scaled Jacobians near 1.0, corresponding to orthogonal corners in a mesh.

Canann et al. [29] use local smoothing operations to optimize the elements' shape in a given mesh. The process can be guided by a distortion metric, where one based on the minimal angle for quad meshes is offered. Garimella et al. [30] improve the quality of a given mesh based on local parametrization. The proposed objective function is based on squared edge lengths. The intention is to smooth distribution of edge lengths and face angles around a vertex, where each pair of edges tries to reach equal length and form a right angle. Zhang et al. [24] improve the quality of quad (and hex) meshes using geometric flow. Surface diffusion flow is performed to remove noise by relocating vertices in the normal direction, and the aspect ratio is improved while preserving features by adjusting vertex positions in tangent directions. One of the quality measures of the result is the scaled Jacobian. In [31], the shape of facets of a given quad mesh is improved using local operations. These are performed on individual elements, followed by a global projection step that reconciles the new vertex positions and assembles the elements into a continuous mesh. Xu et al. [32] suggest a technique to untangle and improve a hex mesh, aiming to increase the minimum scaled Jacobian. To improve the element quality, an angle-based optimization strategy is proposed, which is based on shear energy. To handle non-linearity, a local-global strategy is utilized.

The methods in the previous paragraph operate on a given mesh connectivity without changing it. Moreover, mapping back the adjusted vertices to the original surface is not necessarily performed. In contrast, we improve a seamless mapping before a quad mesh is extracted from it, thus affecting the construction and topology of the resultant quad mesh, whose vertices lie on the original surface.

In garment manufacturing, excessive shear is undesirable, even if physically possible, because it leads to wrinkles and impression of bad fit. Pietroni et al. [33] present a real-time application for garment tailoring. They note that the shear measure is non-quadratic, which is not suited for real-time optimization. Therefore, they settle for a common isometry measure instead. McCartney et al. [34, 35] consider shear distortion in the warp direction only in a woven fabric.

**Anisotropy.** Section 5 offers an approach to produce anisotropic mappings with reduced shear. We review some work related to anisotropic, seamless mappings. Bommes et al. [1] offer to create an anisotropic mapping by scaling the frames in the layout stage. The scaling can be user-defined or derived from curvature. Note that while giving more weight to a specific direction biases the energy, the energy remains oblivious to shear. In [6], sparse user-defined anisotropic scaling is smoothly interpolated to the whole mesh. Anisotropy is dealt with by deforming the mesh itself while proceeding with the common cross field pipeline. Kovacs et al. [36] use a metric derived from the shape operator to derive the anisotropic scaling, with consideration to the approximation error. In contrast to these methods, we allow arbitrary anisotropic scaling, where the scaling amount is optimized to reduce shear distortion.

### 3. Background

Let  $\mathcal{M} = (\mathcal{V}; \mathcal{E}; \mathcal{T})$  be a surface triangle mesh (consisting of vertices, edges, and triangles). Let  $\tilde{f}: \mathcal{M} \subset \mathbb{R}^3 \rightarrow \mathbb{R}^2$  be a piecewise linear surface parametrization that maps a point  $p \in M$  to the  $UV$ -plane,  $\tilde{f}(p) = (u, v)$ .

$\tilde{f}(p)$  is composed of  $\tilde{f}(p) = f(Rp)$ , where  $R \in \mathbb{R}^{2 \times 3}$  is an arbitrary matrix that maps a triangle  $t \in \mathcal{T}$  (containing  $p$ ) isometrically to a *reference XY*-plane, and  $f: \mathbb{R}^2 \rightarrow \mathbb{R}^2$  that maps vectors from the *XY*-plane to the *UV*-plane. From here on out, we omit the step of mapping to and from the reference plane and refer to  $f$  as the mapping from the surface to the *UV*-plane.

Given a mapping  $f$  of a triangle  $t \in \mathcal{T}$ , let  $\sigma_1 \geq \sigma_2$  be the (*signed*) singular values of the mapping Jacobian  $J = Df$ . The singular values indicate the minimum and maximum amount of scaling that is applied to a mapped vector. They are used in popular distortion measures.

The isometric distortion measure that we use is defined as in [37]:

$$\tau(f) = \begin{cases} \max\left(\sigma_1, \frac{1}{\sigma_2}\right) & \sigma_2 > 0 \\ \infty & \text{else} \end{cases}. \quad (1)$$

Given the large dilatation  $K(f) = \frac{\sigma_1}{\sigma_2}$ , the conformal distortion measure that we use is the little dilatation [38]:

$$k(f) = \frac{K - 1}{K + 1}. \quad (2)$$

We use two popular energies to minimize these distortion measures: the isometric symmetric Dirichlet energy [39, 40, 41]

$$E_{iso}(f) = \sigma_1^2 + \frac{1}{\sigma_1^2} + \sigma_2^2 + \frac{1}{\sigma_2^2}, \quad (3)$$

and the conformal MIPS energy [42]

$$E_{conf}(f) = \frac{\sigma_1}{\sigma_2} + \frac{\sigma_2}{\sigma_1}. \quad (4)$$

The common field-based seamless parametrization pipeline consists of two steps [1]. In the first step, a guiding cross field is found, and cone singularities are identified. In the second step, the surface is cut along the singularities, and the surface is laid out, optimizing a mapping  $f$  such that its Jacobian strives to align with frames extracted from the cross field. Given a frame  $A$ , a common alignment energy [1] is

$$\bar{E}_{frame}(f) = \|J - A\|_F^2. \quad (5)$$

We use a symmetric version of this energy as an energy and a measure of aligning the Jacobian with a frame:

$$E_{frame}(f) = \xi(J) = \frac{1}{2} (\|J - A\|_F^2 + \|J^{-1} - A^{-1}\|_F^2). \quad (6)$$

A frame-alignment energy is usually used to generate an initial injective mapping. If the frames are orthogonal (e.g. from a cross field), then optimizing field alignment promotes isometry. Further optimization of the mapping is performed through minimizing energies such as Equations (3) and (4).

For a more detailed background on seamless parametrization, how it differs from integer seamless parametrization (additional integral constraints on cone positions and twin edge translations), and the pipeline for employing it for quad mesh generation, we refer the reader to [1, 2, 3].

#### 4. Shear Energy

While in the first step a cross field with perfect orthogonal unit vectors is constructed, the final distortion is determined in the second layout step where the mapping is optimized. Minimizing common energies such as Equations (3) and (4) does not penalize shearing. In this section, we describe a specific shear energy to augment these energies. But first, we elaborate some more on the motivation.

In most cases, there must be isometric or conformal distortion, and often it can be high. Our proposed rationale is then that when, e.g., isometry has to be given up, we might as well pay an additional small increase in the overall isometric distortion to significantly lower shear distortion (at the expense of anisotropy). At least, in terms of visualization, the effect could be enormous.

A note about the intended edge flow. Constructing the cross field in the first step determines the cone singularities, which in turn, determine the basic structure of the edge flow. That is, the second step of optimizing the mapping does not affect the edge flow dramatically. Therefore, it is safe for us to do the best we can in terms of optimization (adhering to sharp-edge constraints if present) without compromising the user's intention (if there is such) regarding the edge flow.

We define the shear energy as follows. Let  $e_1, e_2 \in \mathbb{R}^2$  be the unit vectors of the  $U$ - and  $V$ -axis (columns of the  $2 \times 2$  identity matrix). Let  $\hat{j}_i = J^{-1}e_i \in \mathbb{R}^{2 \times 1}$  be the mapped axis vectors from the domain to the surface (i.e. the

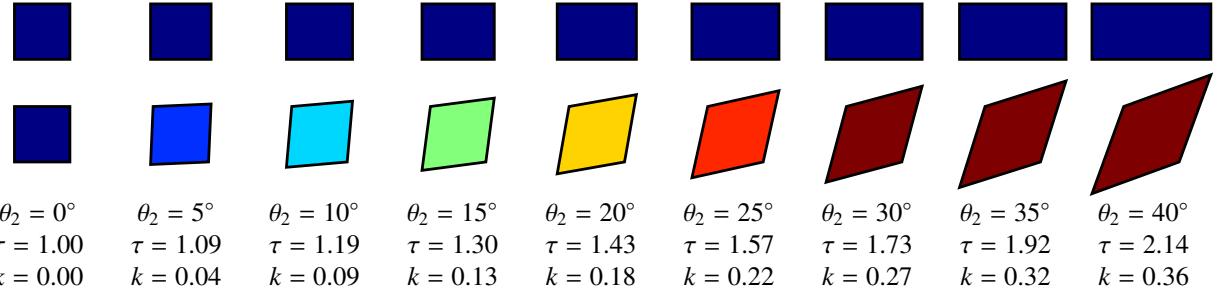


Figure 3. Distortion illustration. Similar to the settings in Figure 2, each shape is the result of a square in a checkerboard pattern being pulled back to the surface via the parametrization. The inverse Jacobian of the mapping is composed of  $J^{-1} = R\bar{\Sigma}R$ , where  $R$  is a rotation, and  $\bar{\Sigma}$  is a scaling transformation defined in Section 4.2. In the first row,  $R$  is the identity, and in the second row,  $R$  is a  $45^\circ$  rotation. The singular values for a rectangle mapping are the rectangle's width and height. Each column contains the same amount of isometric and conformal distortion,  $\tau$  and  $k$ . The first row of rectangles has  $\theta_1 = 0^\circ$  shear distortion. The shear distortion of the second row is given under it. The shear distortion for a given  $k$  can vary from  $0^\circ$  to  $\theta_2$  (Section 4.2). The colors correspond to the shear distortion color bar in the figures.

columns of  $J^{-1}$ ). To preserve the right angles of a checkerboard pattern that is mapped back to the surface, we need to keep the orthogonality of the mapped axis vectors. This leads to the shear distortion measure  $\theta(f)$ :

$$\zeta(f) = \arccos \frac{\hat{j}_1 \cdot \hat{j}_2}{\|\hat{j}_1\| \|\hat{j}_2\|}$$

$$\theta(f) = |90 - \zeta|, \quad (7)$$

which measures the difference between  $90^\circ$  and the angle  $\zeta$  between the mapped axis vectors (the *shear-defect*). This encourages the scaling of a deformed quad to be along the axes and preserve right angles. For illustration, see Figure 3. In this work, degrees are used as the measurement unit for angles (and trigonometric functions).

We now define the shear energy. Let  $M(J) = JJ'$  (which is not the same as the metric tensor  $J'J$ ), and define the entries of the symmetric matrices

$$M = \begin{bmatrix} m_{11} & m_{12} \\ m_{12} & m_{22} \end{bmatrix}, \quad M^{-1} = \begin{bmatrix} \hat{m}_{11} & \hat{m}_{12} \\ \hat{m}_{12} & \hat{m}_{22} \end{bmatrix} = \frac{1}{\det M} \begin{bmatrix} m_{22} & -m_{12} \\ -m_{12} & m_{11} \end{bmatrix},$$

where  $m_{ij} = \text{tr } e'_i M e_j = \text{tr } E_{ij} M$ ,  $E_{ij} := e_j e_i'$  (the trace is invariant to cyclic permutation, and a scalar is equal to its trace). Next, define the function  $\phi(J^{-1}) : \mathbb{R}^{2 \times 2} \rightarrow \mathbb{R}$ :

$$\phi(J^{-1}) = \left( \frac{\hat{j}_1 \cdot \hat{j}_2}{\|\hat{j}_1\| \|\hat{j}_2\|} \right)^2 = \frac{\hat{m}_{12}^2}{\hat{m}_{11} \hat{m}_{22}},$$

which is based on Equation (7), and using  $\hat{j}_i \cdot \hat{j}_j = \text{tr}(J^{-1} e_i)' J^{-1} e_j = \text{tr} E_{ij} M^{-1}$ .  $\phi$  can also be viewed as measuring the (normalized) off-diagonal entries of  $M^{-1}$ . The shear energy is defined as

$$E_{\text{shear}}(f) = \phi(J^{-1}). \quad (8)$$

The energy is scale-invariant (referring to a global scale of the mesh; see Section 6 for a discussion about the advantage of a scale-invariant energy), and since it is not rotation-invariant (the orientation of a domain triangle), it cannot be expressed purely with singular values (unlike conformal and isometric energies [43]). Scale-invariance implies that the energy is invariant to  $\det M$ , which leads to being invariant to inverting  $M$ :

$$\phi(J^{-1}) = \frac{m_{12}^2}{m_{11} m_{22}}.$$

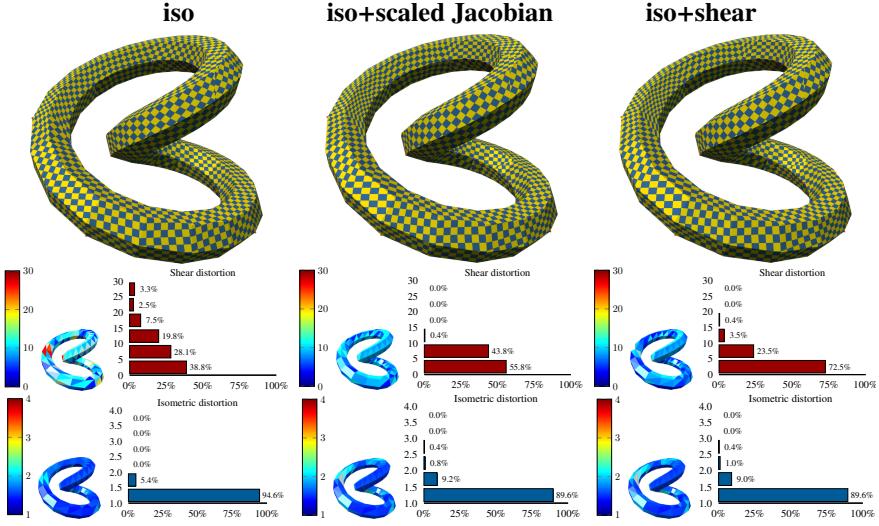


Figure 4. A comparison of an energy based on the scaled Jacobian and our shear energy. In the experiment, each energy was minimized with a large weight in the objective, and a specific upper bound on  $E_{iso}$ . For that, we used a general purpose constrained solver. The average shear distortion  $\theta$  (weighted by area) of  $E_{shear}$  (3.6) was lower than  $E_{jac}$  (4.5), and both were lower than  $E_{iso}$  minimized alone (8.2). Labels: 'iso'— $E_{iso}$ , 'scaled Jacobian'— $E_{jac}$ , 'shear'— $E_{shear}$ . Distortion measures:  $\theta$  and  $\tau$ .

#### 4.1. The Relation to the Scaled Jacobian Measure

The popular scaled Jacobian measure can be expressed for a planar mapping as:

$$\frac{|J^{-1}|}{\|\hat{j}_1\| \|\hat{j}_2\|} = \frac{\hat{j}_1 \times \hat{j}_2}{\|\hat{j}_1\| \|\hat{j}_2\|} = \sin \zeta, \quad (9)$$

where  $|J^{-1}|$  is the determinant of the Jacobian of the inverse mapping. An energy based on this measure, for example:

$$E_{jac}(f) = \left(1 - \frac{|J^{-1}|}{\|\hat{j}_1\| \|\hat{j}_2\|}\right)^2. \quad (10)$$

The relation between  $E_{shear}$  and  $E_{jac}$  is that the former is based on  $\cos \zeta$  while the latter is based on  $\sin \zeta$ . They both measure the amount of shear. Another possible energy is:

$$\bar{E}_{jac}(f) = 1 - \left(\frac{|J^{-1}|}{\|\hat{j}_1\| \|\hat{j}_2\|}\right)^2, \quad (11)$$

which is equal to  $E_{shear}$ .

One of the conclusions in the measure study [12] is that a skewness metric that is based on  $\cos \zeta$  performs better than the scaled Jacobian that is based on  $\sin \zeta$ . For a comparison, see Figure 4.

#### 4.2. The Relation to the Singular Values

To relate the conformal distortion to the bound on the shear distortion, consider the singular value decomposition of the mapping Jacobian and the decomposition of its inverse

$$J = U\Sigma V', \quad J^{-1} = V'\Sigma^{-1}U, \quad \Sigma = \begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix},$$

which provide the view of  $J^{-1}$  as a composition of two rotations and a scaling transformation.

Since the conformal and shear distortions are invariant to scaling, we can scale the mapping by  $\sigma_2^{-1}$ , which would replace the scaling matrix in the decomposition of  $J^{-1}$  with  $\bar{\Sigma}^{-1} = \begin{bmatrix} K^{-1} & 0 \\ 0 & 1 \end{bmatrix}$  (and the isometric distortion would be  $\tau = K$ ).

If  $U$  is the identity, then there would be no shearing. If  $U$  is a  $45^\circ$  rotation, then the amount of shearing would be maximal. The rotation  $V$  has no effect on the shearing. Given a large dilatation  $K$ , the maximal shear distortion is

$$\theta_{max} = 90^\circ - 2 \operatorname{arccot} K .$$

See Figure 3 for illustration.

For an isometric distortion  $\tau$ , the maximal conformal distortion is achieved when  $\tau = \sigma_1 = \sigma_2^{-1}$ , and the large dilatation in this case is  $K = \tau^2$ .

## 5. Smoothness Energy

As an alternative regularization energy (to e.g. isometric and conformal energies), we offer a second-order smoothness energy that strives to keep the Jacobian function smooth.

Let  $f_1$  and  $f_2$  be mappings of two faces that share an edge (on the surface), with corresponding Jacobians  $J_1$  and  $J_2$  (transformed to the same coordinate system; see Appendix C). Define:

$$\eta(J_1, J_2) = \|J_1 - J_2\|_F^2$$

$$\psi(J_1, J_2) = \frac{\eta}{|J_1|} + \frac{\eta}{|J_2|} ,$$

where  $|J_i|$  is the determinant of  $J_i$ . The smoothness energy of a mapping  $f_i$  of  $t_i \in \mathcal{T}$  is

$$E_{smooth}(f_i) = \sum_{j \in ne(i)} \psi(J_i, J_j) , \quad (12)$$

where  $ne(i)$  is a set of indices of faces that share an edge with  $t_i$ . This energy is scale-invariant (but not rotation-invariant). Scale-invariance is not common in smoothness energies [44, 45], and this form of normalization prevents the common shrinking and vanishing of the mapping.

The combination of the smoothness and shear energies furnishes an anisotropic energy, which is usually more flexible (when smoothness permits) than a conformal energy. Anisotropic elements have been known to have better approximation quality [36]. Here, their flexibility produces smooth mappings with less shear.

## 6. Optimization

The energies are integrated over the mesh, using triangle areas as weights, and normalized by the total surface area:

$$E^M(f) = \frac{1}{A_M} \sum_{t \in \mathcal{T}} A_t \left( \lambda E_{shear}(f_t) + E_{reg}(f_t) \right) , \quad (13)$$

where  $f_t$  is the mapping over a triangle  $t$ ,  $A_t$  is  $t$ 's area, and  $A_M$  is the total surface area of the mesh.  $E_{reg}$  is one of the regularization energies:  $E_{iso}$ ,  $E_{conf}$ , and  $E_{smooth}$ .  $\lambda$  is a weighing scalar.

The energies, except for  $E_{iso}$ , are scale-invariant. That is, if  $\hat{\mathcal{M}}$  is a globally-scaled version of  $\mathcal{M}$  by a factor  $\gamma$  (i.e.  $\hat{\mathcal{V}} = \gamma \mathcal{V}$ ) with the domain remaining intact, i.e.  $\hat{f}(\hat{\mathcal{M}}) = f(\mathcal{M})$ , then  $E(\hat{f}) = E(f)$ . This is not case for  $E_{iso}$ .  $E_{iso}$ , however, is *optimal-scale-invariant*: a global scaling of  $\mathcal{M}$  does not affect the value of the energy of the optimal mapping. Meaning, if  $f^*$  is a minimizer of  $E_{iso}$ , then  $E_{iso}(\hat{f}^*) = E_{iso}(f^*)$ . Moreover,  $\hat{f}^*(\hat{\mathcal{M}}) = \gamma f^*(\mathcal{M})$ . That is, the mapping will be scaled “automatically” to reach optimality. Optimal-scale invariance is a desirable property in an energy. It allows the user to use a weighing factor to weigh between optimal-scale-invariant energies without the necessity to adjust it to the mesh size. In our case, we used  $\lambda = 10$  in all our experiments. In terms of optimization, scale-invariant energies do not require special handling when optimizing, as detailed next.

$E^M$  is minimized subject to linear seamlessness constraints [1]. For that, we used the projected Newton method [46, 43], augmented with the foldover-free line search proposed in [39]. This line search method prevents inversion of a triangle (preserves orientation) by starting the search before a triangle collapses (or inverts). For our scale-invariant energies, or more generally, an energy that contains division by  $\sigma_2$  (or related expressions such as  $\det J$ ), nearly collapsed triangles are heavily penalized. As a consequence, starting from an initial locally injective mapping, we are guaranteed that the mapping remains locally injective throughout the optimization process.

When optimizing a scale-invariant energy, the optimizer does not have a reason to stray far from the global scaling of the initial mapping, and the ratio of total domain area to total surface area is likely to remain the same. Nevertheless, the final scaling would be arbitrary. Therefore, in the results, we globally scale the mappings such that the total mapped area would be equal to the total surface area.

The projected Newton method requires the Jacobian and Hessian of the energies. The derivatives of the singular-value-based energies  $E_{iso}$  and  $E_{conf}$  or their proxies can be found in [43, 40, 41], and the derivatives of  $E_{shear}$ ,  $E_{smooth}$ , and  $E_{frame}$  are given in the appendices.

## 7. Evaluation

We evaluated the method and compared it with the state of the art; see Figures 5 to 10. The dataset is similar to [3]. Note that if a facet is not directly facing the camera, a pattern may look sheared due to perspective projection. Also, when the surface is not flat, a square pattern may look distorted, and judging its angles rather than its general shape is more telling. The shear distortion heat maps provide assistance in these cases.

In general, when  $E_{shear}$  was added to  $E_{iso}$  or  $E_{conf}$ , the amount of shear distortion was decreased substantially, with a reasonably low increase in the amount of isometric or conformal distortion. The effects of lowering the shear distortion were more apparent in places with high isometric or conformal distortion. The visual improvement in the shape of a quad takes place without improving the isometric or conformal distortion.

$E_{smooth}$  was compared with  $E_{conf}$  (which is more flexible than  $E_{iso}$ ). Since the two objectives do not coincide, the conformal measure as a comparison tool is not adequate. On the other hand, for the same reason, measuring smoothness is not a fair comparison either. Bearing that in mind, we opted for using the conformal measure, which is more popular, and it gives an indication of how much conformal distortion the smoothing has introduced.

Quantitative measures aside, the mapping results of  $E_{smooth}$  are visually smoother. On the other hand,  $E_{shear}$  often reduces the mapping smoothness and at times creates discontinuity-like lines in the pattern. For example, see the first blowup in Figure 10, where the vertical feature edges pull the pattern in one direction while the horizontal feature edges pull it in another. The effect of reducing shear in this case is that the pattern appears as if divided along a diagonal line.  $E_{smooth}$  remedies that by sacrificing conformal (and isometric) distortion, which could be considerable.

**Run time.** The experiments were conducted on a laptop with a six-core 2.6GHz CPU and 32GB RAM. Tables 1 and 2 summarize the running time. The run time is dominated by constructing the Hessian and solving the KKT system for a search direction. The optimization requires an initial, feasible point (satisfying seamlessness constraints). We used [1] to acquire a smooth cross field, and we generated the initial layout by employing [37], using the cross field as local frames [47, 3].

In our implementation, when calculating  $E_{shear}$ , the run time increased only in the first step of constructing the Hessian. The run time in the second step of solving the KKT remained the same, since the sparsity pattern of the system matrix remained unchanged. The main factor for the total run time is the number of iterations, which was increased due to the initial mapping that minimized an isometric energy. The same stopping criteria (based on the gradient magnitude and the changes in energy and UV solution) was used in all cases.

$E_{smooth}$  normally adds two variables to a local term for a triangle (the  $UV$  coordinates of an additional neighboring vertex), and it can add up to six variables to triangles near the seam. This increased the run time (per iteration) of the two optimization steps. Moreover, the addition of  $E_{smooth}$  required more iterations to converge, which is reasonable since the initial point was an isometric mapping.

**Field alignment.** We tested our shear energy in conjunction with  $E_{frame}$ . In Figure 11, we generated a cross field that strives to align with principle curvature directions using [1] (including a post-processing step of merging nearby cones; the cost is increased distortion). The shear distortion is reduced while the alignment with principle curvature

model	#tri	iso						iso+shear					
		θ	τ	it	Hess	solve	total	θ	τ	it	Hess	solve	total
armadillo	43160	7.6	1.3	10	1.4	0.5	19	1.6	1.3	15	1.5	0.5	30
armchair	100000	7.4	2.7	10	3.5	0.9	44	0.8	3.0	25	3.6	0.9	112
botijo	82332	8.8	1.3	11	3	0.9	43	1.5	1.4	15	3.3	0.9	63
bunnyBotsch	111364	8.9	1.3	9	5.1	1.1	56	1.3	1.4	9	5.4	1.1	59
dancer_25k	49996	8.7	1.6	11	2.1	0.5	29	1.6	1.5	17	2.2	0.5	46
dilo_scaled	54344	6.8	1.9	14	2.5	0.5	42	1.3	1.9	23	2.6	0.5	72
elephant	49918	7.7	1.6	9	2.3	0.5	25	1.4	1.5	13	2.3	0.5	37
knot100K	100000	7.7	1.3	10	5.2	0.9	61	1.6	1.3	20	5.6	0.9	131
pear	21504	9.5	1.3	8	0.9	0.2	9	1.3	1.4	19	0.9	0.2	21
neptune0	105030	7.7	1.2	13	6	1.1	92	1.4	1.2	14	6.2	1.1	102
torso	100000	9.9	2.0	12	5.5	1	78	1.2	2.1	22	5.7	1	147
uu-memento100k	99932	8.6	1.2	9	5.5	1	59	1.3	1.3	24	5.8	1	162
beetle_refined	38726	4.6	1.3	7	1.3	0.3	11	1.2	1.3	8	1.4	0.3	14
casting_refined	36852	6.9	1.3	9	1.7	0.5	19	1.7	1.3	9	1.6	0.4	19
sculpt	7342	16.8	2.0	8	0.3	0.1	3	4.3	2.2	12	0.3	0.1	5
fandisk	14454	7.2	1.6	8	0.5	0.2	6	2.1	1.6	11	0.6	0.2	9

Table 1. Columns: '#tri'—number of triangles, 'iso'— $E_{iso}$ , 'shear'— $E_{shear}$ ,  $\theta$  and  $\tau$ —distortion measures, 'it'—number of iterations, 'Hess'—average time per iteration to calculate the Hessian, 'solve'—average time per iteration to solve the KKT system for a search direction, 'total'—total time. Time is given in seconds.

model	#tri	MIPS						MIPS+shear						smooth+shear					
		θ	k	it	Hess	solve	total	θ	k	it	Hess	solve	total	θ	k	it	Hess	solve	total
cup	11340	2.7	0.04	11	0.4	0.2	7	0.9	0.05	14	0.5	0.2	10	0.8	0.27	59	0.8	0.2	59
linkCupTop	19560	3.7	0.05	13	0.8	0.3	14	2.3	0.06	25	0.9	0.3	30	2.5	0.14	55	1.3	0.3	88
octocat-v1	37884	3.7	0.05	9	1.8	0.3	19	2.0	0.06	12	1.9	0.3	26	2.8	0.12	43	2.7	0.3	129
oni	2845	3.6	0.05	8	0.1	0.1	2	1.7	0.06	11	0.2	0.1	3	2.4	0.15	17	0.2	0.1	5
genus3	13312	4.4	0.06	9	0.5	0.2	6	1.4	0.08	17	0.6	0.2	14	1.0	0.19	30	0.9	0.2	33
helmet	1000	4.3	0.06	9	0.1	0	1	1.8	0.07	12	0.2	0	2	2.3	0.18	24	0.2	0	5
fandisk	14454	6.1	0.12	8	0.5	0.2	6	1.5	0.12	6	0.6	0.2	5	1.1	0.26	16	0.9	0.2	18

Table 2. Columns: '#tri'—number of triangles, 'MIPS'— $E_{conf}$ , 'shear'— $E_{shear}$ , 'smooth'— $E_{smooth}$ ,  $\theta$  and  $k$ —distortion measures, 'it'—number of iterations, 'Hess'—average time per iteration to calculate the Hessian, 'solve'—average time per iteration to solve the KKT system for a search direction, 'total'—total time. Time is given in seconds.

directions is (mostly) preserved. The chair example illustrates that even when the distortion is low, there is still room for improvement. Minimizing this energy combination, we can optimize any given mapping to reduce its shear. For that, we use the mapping Jacobian as target frames (using polar decomposition to retrieve closest orthogonal frames).

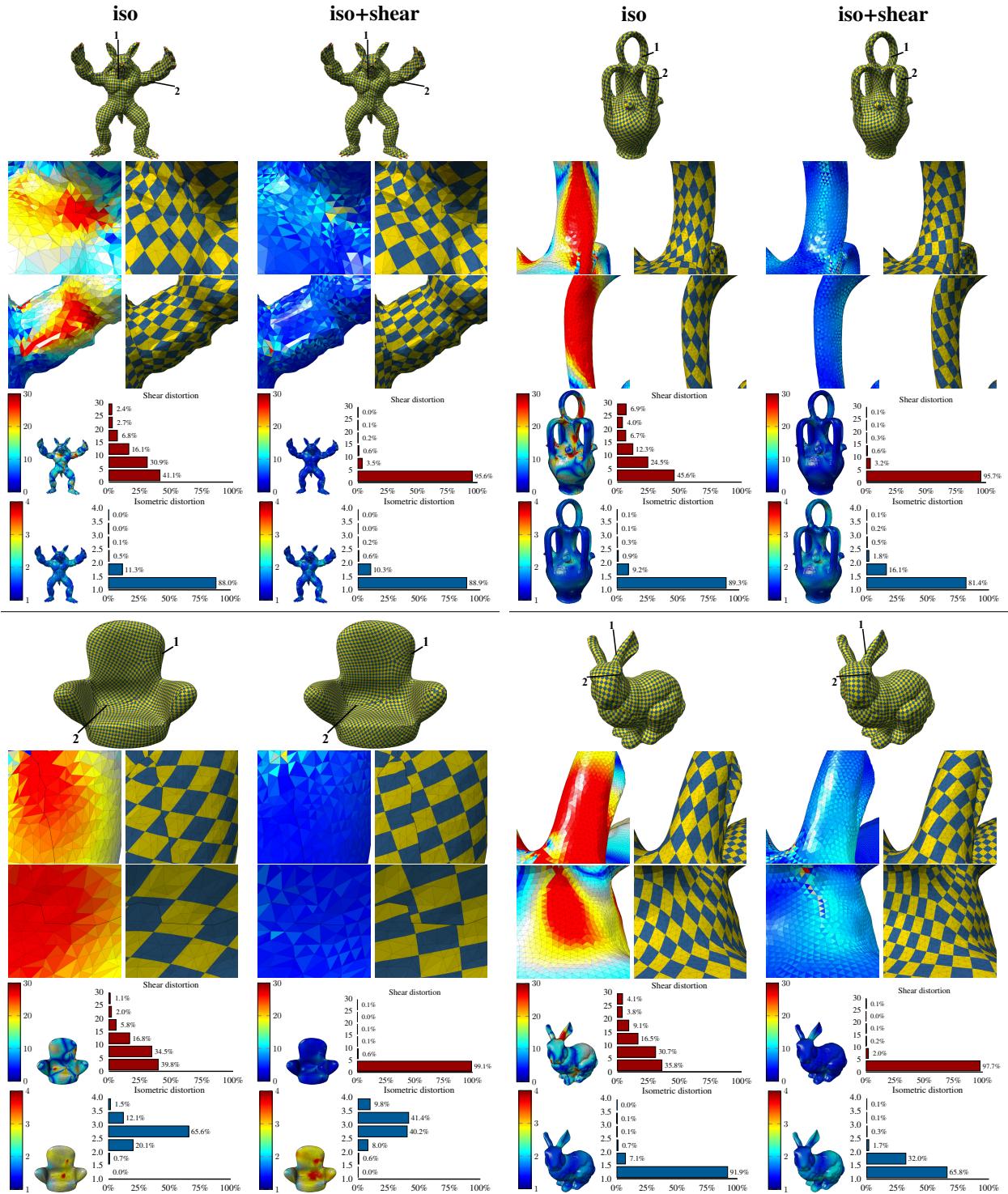
## 8. Conclusion

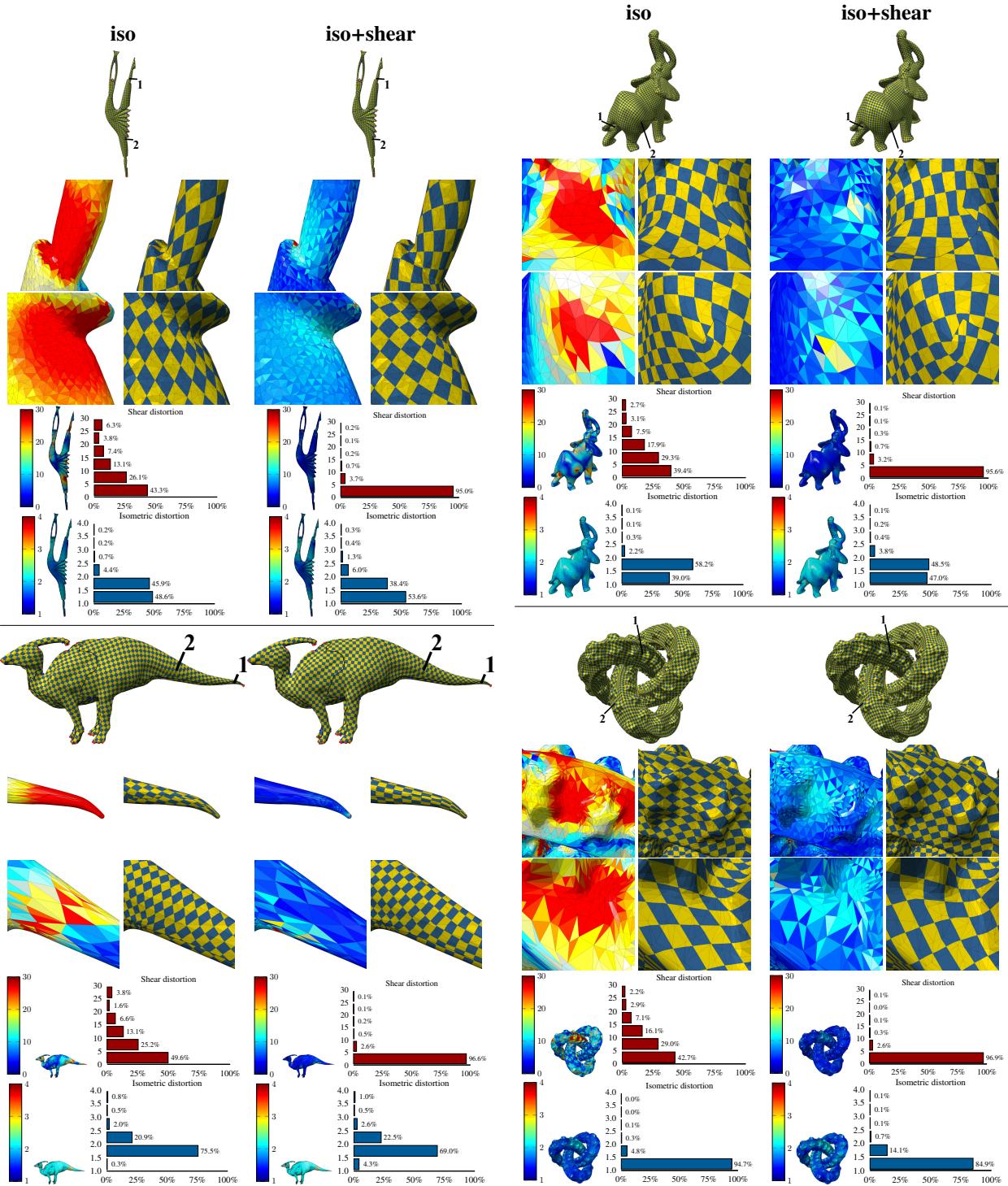
We pointed out an oversight of well-known energies in regard to shear distortion and introduced a new shear energy to address it. We also introduced a scale-invariant, second-order smoothness energy that leads to anisotropic elements. Both energies were incorporated efficiently into the latest foldover-free optimization framework alongside popular isometric and conformal energies. Efficiency of the algorithm and the quality of the resultant mapping were demonstrated through experiments, along with comparison to the state of the art.

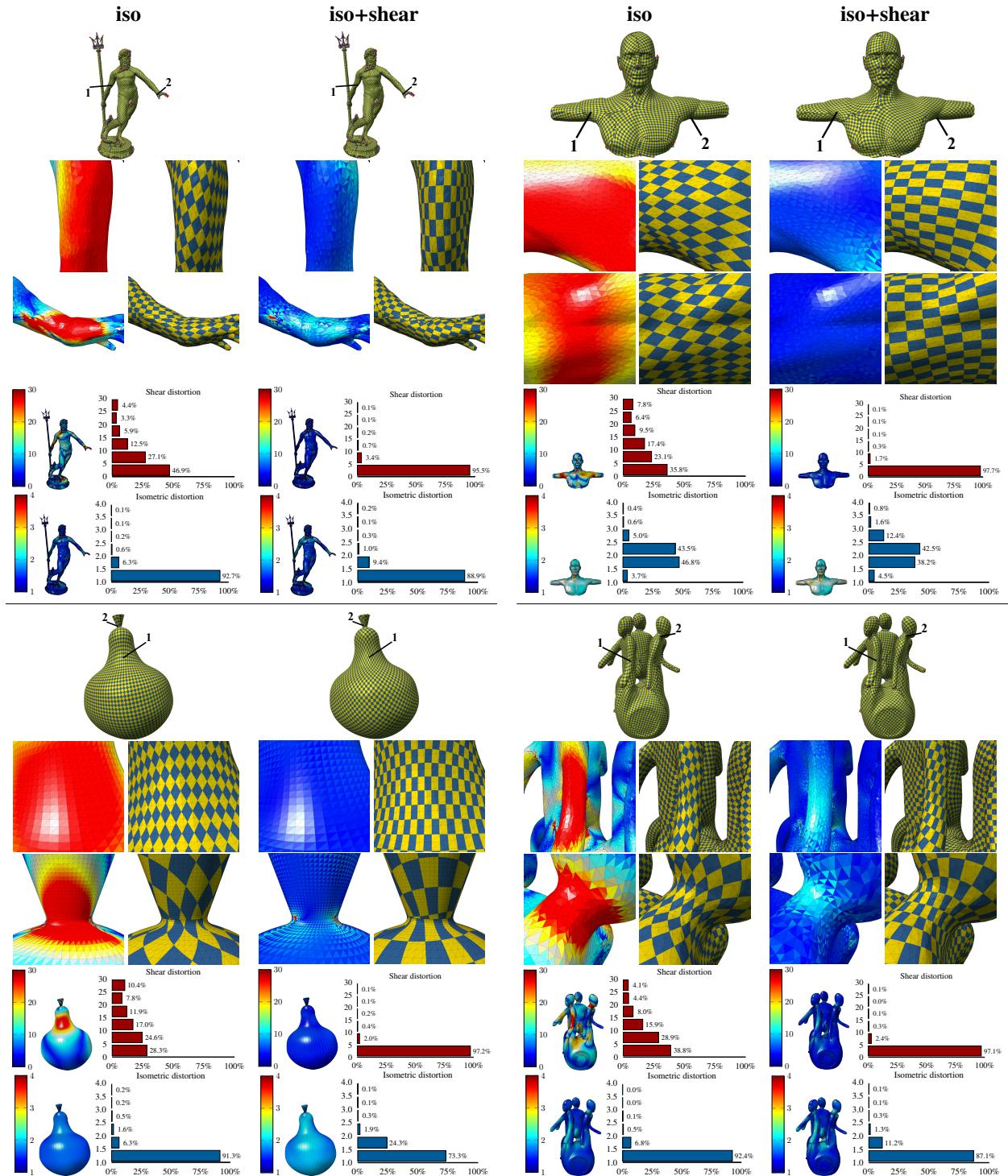
As an avenue for future work, it would be interesting to generalize the energy and the concepts explored in this paper to tetrahedral and hex meshes.

## Acknowledgments

We would like to acknowledge the anonymous reviewer who pointed out that Equation (8) is invariant to inverting  $M$ .

Figure 5. Labels: 'iso'— $E_{iso}$ , 'shear'— $E_{shear}$ . Distortion measures:  $\theta$  and  $\tau$ . The blowup heat maps show  $\theta$ .

Figure 6. Labels: 'iso'— $E_{iso}$ , 'shear'— $E_{shear}$ . Distortion measures:  $\theta$  and  $\tau$ . The blowup heat maps show  $\theta$ .

Figure 7. Labels: 'iso'— $E_{iso}$ , 'shear'— $E_{shear}$ . Distortion measures:  $\theta$  and  $\tau$ . The blowup heat maps show  $\theta$ .

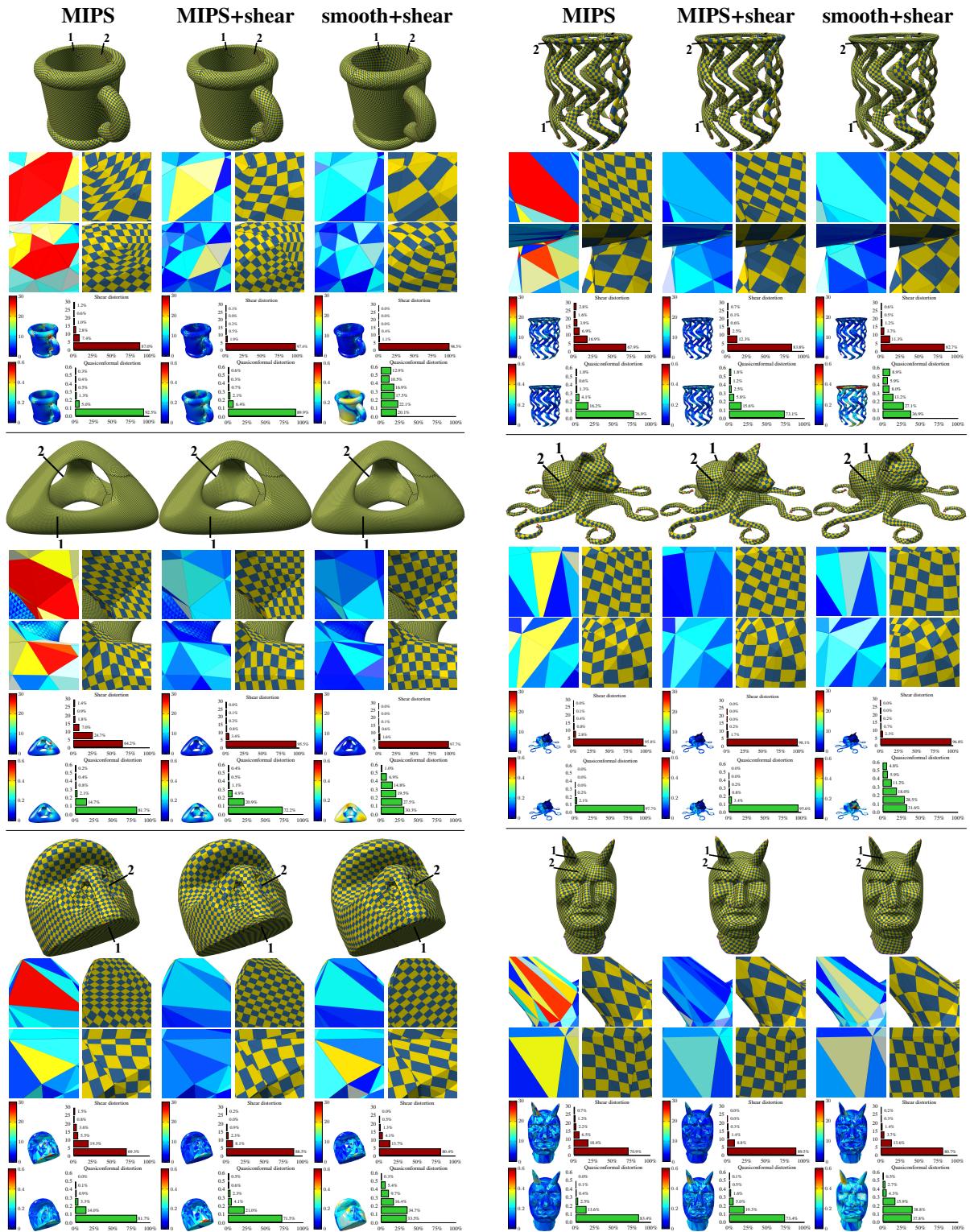


Figure 8. Labels: 'MIPS'— $E_{conf}$ , 'shear'— $E_{shear}$ , 'smooth'— $E_{smooth}$ . Distortion measures:  $\theta$  and  $k$ . The blowup heat maps show  $\theta$ .

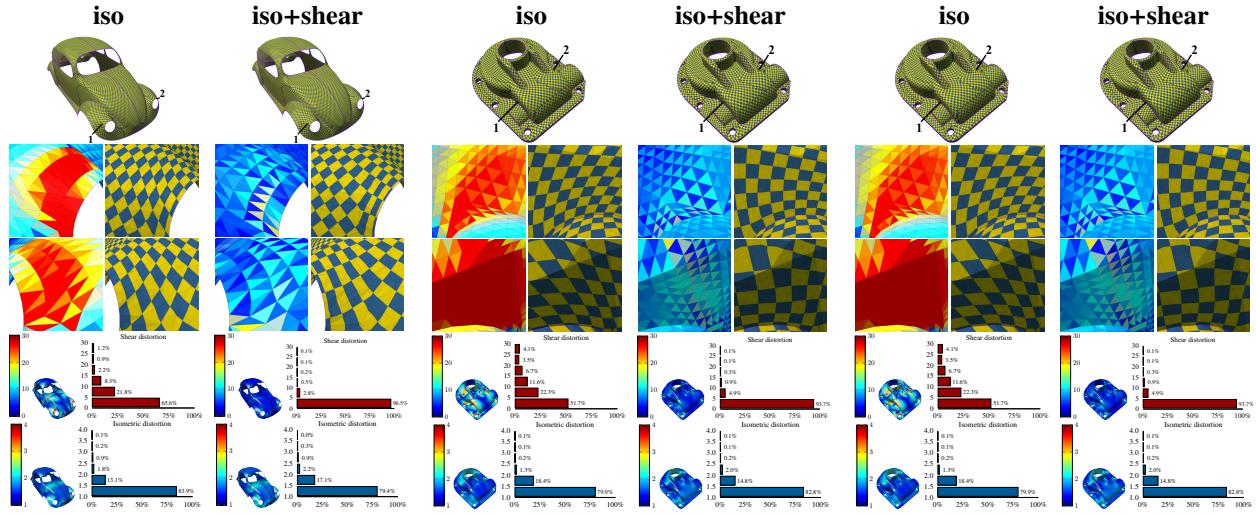


Figure 9. Sharp features. Labels: 'iso'— $E_{iso}$ , 'shear'— $E_{shear}$ . Distortion measures:  $\theta$  and  $\tau$ . The blowup heat maps show  $\theta$ .

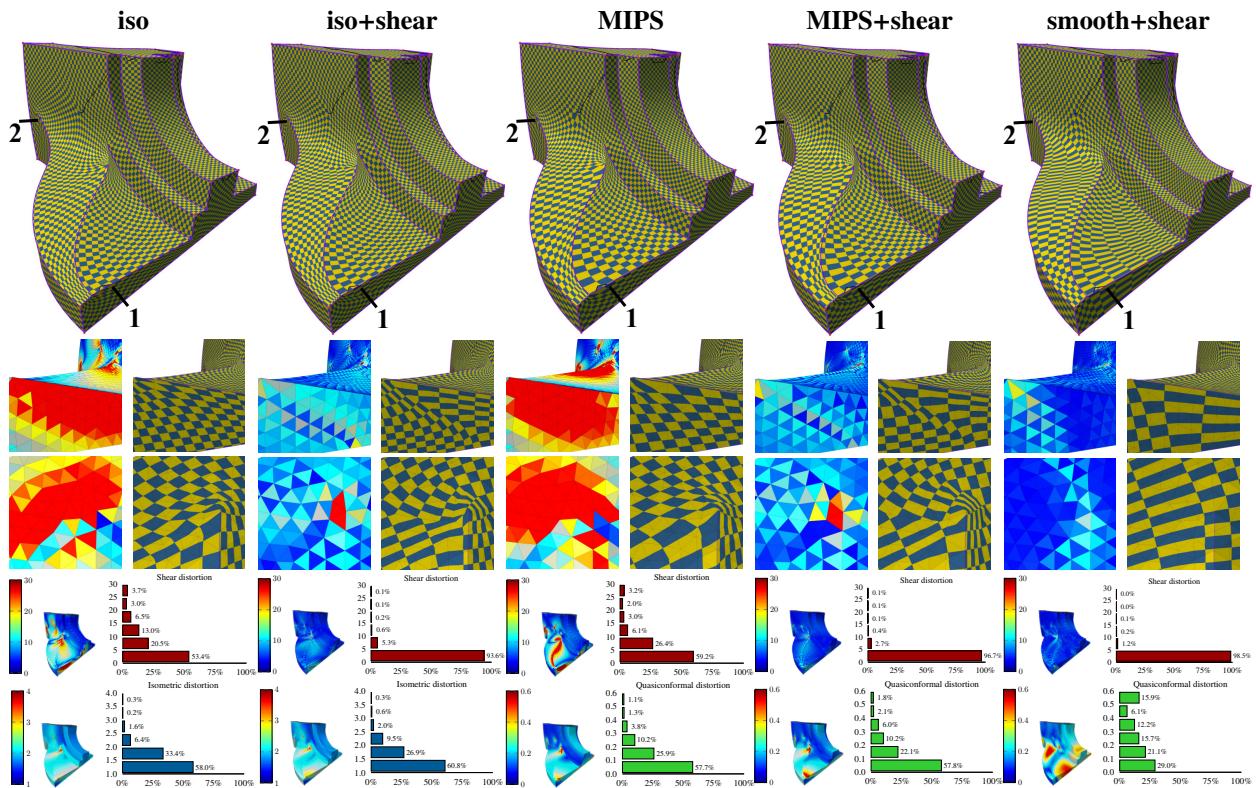
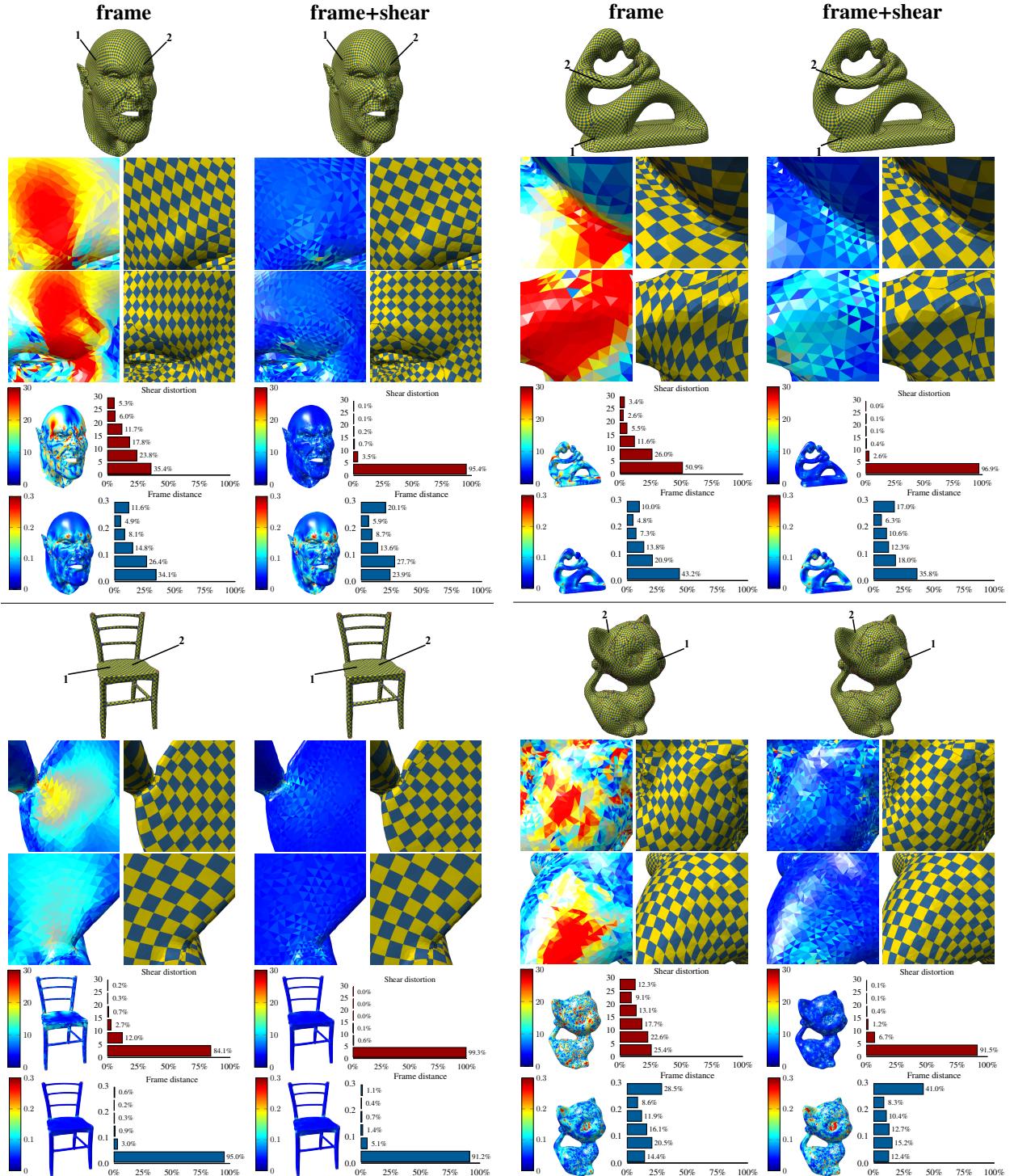


Figure 10. Sharp features. Labels: 'iso'— $E_{iso}$ , 'MIPS'— $E_{conf}$ , 'shear'— $E_{shear}$ , 'smooth'— $E_{smooth}$ . Distortion measures:  $\theta$ ,  $\tau$ , and  $k$ . The blowup heat maps show  $\theta$ .

Figure 11. Labels: 'frame'— $E_{frame}$ , 'shear'— $E_{shear}$ . Distortion measures:  $\theta$  and  $\xi$ . The blowup heat maps show  $\theta$ .

## Appendix A. Differentiation of the Mapping Jacobian

In the following, we use notations and definitions from [48]. In each derivative subsection, we begin by differentiating an inner function and proceed with differentiating functions along a function composition chain. Finally, we use the chain rule to calculate the derivative of the main (outer) function with respect to  $X$ .

We start by differentiating the mapping Jacobian, which is used later when differentiating the energies.

Let  $X \in \mathbb{R}^{3 \times 2}$  be three rows of vertex  $UV$  coordinates of a mapping  $f$  of  $t \in \mathcal{T}$ . The Jacobian  $J \in \mathbb{R}^{2 \times 2}$  of  $f$  is

$$J(X) = Q(X)P^{-1} = (GX)',$$

where  $P, Q \in \mathbb{R}^{2 \times 2}$  comprise two (column) vector edges of the source and target triangle.  $G \in \mathbb{R}^{2 \times 3}$  is a discrete FEM differential operator [49]. The Jacobian of  $J(X)$  with respect to (a vectorization of)  $X$  is (the constant)

$$\mathbf{D} J(X) = (G \otimes I_2)K_{32} \in \mathbb{R}^{4 \times 6},$$

where  $K_{mn} \in \mathbb{R}^{m \cdot n \times m \cdot n}$  is a commutation matrix,  $I_d \in \mathbb{R}^{d \times d}$  is the identity matrix, and  $\otimes$  is the Kronecker product. The Hessian is a matrix of zeros

$$\mathbf{H} J(X) = \emptyset.$$

## Appendix B. Shear Energy Differentiation

Let

$$m(M) = \begin{bmatrix} m_{11} \\ m_{12} \\ m_{22} \end{bmatrix}_{3 \times 1},$$

where  $m_{ij}$  are the entries of  $M(J) = JJ'$ .

The shear energy is

$$\phi(m) = \frac{m_{12}^2}{m_{11}m_{22}} \in \mathbb{R}.$$

### Appendix B.1. First Derivative

$$\mathbf{D} M(J) = 2N_2(J \otimes I_2) \in \mathbb{R}^{4 \times 4},$$

where  $N_2 \in \mathbb{R}^{4 \times 4}$  is a symmetric idempotent matrix.

$$\mathbf{D} M(X) = (\mathbf{D} M(J)) \cdot (\mathbf{D} J(X)) \in \mathbb{R}^{4 \times 6}$$

$$\mathbf{D} m_{ij}(M) = \left( \text{vec}(E_{ij})' \right)' \in \mathbb{R}^{1 \times 4},$$

where  $E_{ij} = e_j e_i'$ , denoting by  $e_i$  and  $e_j$  the  $i$ th and  $j$ th column of the identity matrix  $I_2$ .  $\text{vec}$  is a vectorization operator, returning a column stack of a matrix.

$$\mathbf{D} m(M) = \begin{bmatrix} \mathbf{D} m_{11}(M) \\ \mathbf{D} m_{12}(M) \\ \mathbf{D} m_{22}(M) \end{bmatrix}_{3 \times 4}$$

$$\mathbf{D} m(X) = (\mathbf{D} m(M)) \cdot \mathbf{D} M(X) \in \mathbb{R}^{3 \times 6}$$

$$\mathbf{D} \phi(m) = \begin{bmatrix} -m_{12}^2 & 2m_{12} & -m_{12}^2 \\ \left( m_{11}m_{22} \right) & \left( m_{11}m_{22} \right) & \left( m_{11}m_{22} \right) \end{bmatrix}_{1 \times 3}.$$

The Jacobian of  $\phi$  with respect to  $X$  is

$$\mathbf{D} \phi(X) = (\mathbf{D} \phi(m)) \cdot \mathbf{D} m(X) \in \mathbb{R}^{1 \times 6}. \quad (\text{B.1})$$

### Appendix B.2. Second Derivative

Define the component function operating on a square matrix  $A \in \mathbb{R}^{2 \times 2}$ :

$$c_{ij}(A) = \text{tr } E_{ij} A \in \mathbb{R}.$$

Then,

$$\mathbf{H} c_{ij}(J) = I_2 \otimes 2E_{ij} \in \mathbb{R}^{4 \times 4}$$

$$\mathbf{H} M(J) = \text{sym} \begin{bmatrix} \mathbf{H} c_{11}(J) \\ \mathbf{H} c_{21}(J) \\ \mathbf{H} c_{12}(J) \\ \mathbf{H} c_{22}(J) \end{bmatrix}_{16 \times 4},$$

where  $\text{sym}$  is a block symmetrization operator:  $\text{sym } B = \frac{1}{2} (B + (B')_v)$  for block matrices  $B, (B')_v \in \mathbb{R}^{mn \times n}$ , consisting of  $B_i \in \mathbb{R}^{n \times n}$ :

$$B = \begin{bmatrix} B_1 \\ B_2 \\ \vdots \\ B_m \end{bmatrix}, \quad (B')_v = \begin{bmatrix} B'_1 \\ B'_2 \\ \vdots \\ B'_m \end{bmatrix}.$$

We proceed with

$$\mathbf{H} M(X) = (I_4 \otimes \mathbf{D} J(X))' \cdot (\mathbf{H} M(J)) \cdot \mathbf{D} J(X) \in \mathbb{R}^{24 \times 6}$$

$$\mathbf{H} \phi(m) = \begin{bmatrix} \frac{2 m_{12}^2}{m_{11}^3 m_{22}} & -\frac{2 m_{12}}{m_{11}^2 m_{22}} & \frac{m_{12}^2}{m_{11}^2 m_{22}^2} \\ -\frac{2 m_{12}}{m_{11}^2 m_{22}} & \frac{2}{m_{11} m_{22}} & -\frac{2 m_{12}}{m_{11} m_{22}^2} \\ \frac{m_{12}^2}{m_{11}^2 m_{22}^2} & -\frac{2 m_{12}}{m_{11} m_{22}^2} & \frac{2 m_{12}^2}{m_{11} m_{22}^3} \end{bmatrix}_{3 \times 3}$$

$$\mathbf{H} m(X) = (\mathbf{D} m(M) \otimes I_6) \cdot \mathbf{H} M(X) \in \mathbb{R}^{18 \times 6}.$$

The Hessian of  $\phi$  with respect to  $X$  is

$$\mathbf{H} \phi(X) = (\mathbf{D} m(X))' \cdot (\mathbf{H} \phi(m)) \cdot (\mathbf{D} m(X)) + (\mathbf{D} \phi(m) \otimes I_6) \cdot \mathbf{H} m(X) \in \mathbb{R}^{6 \times 6}. \quad (\text{B.2})$$

### Appendix C. Smoothness Energy Differentiation

We apply the energy to a triangle hinge of two faces  $t_1, t_2 \in \mathcal{T}$ , consisting of  $n \in \{4, 5, 6\}$  domain vertex copies in total. The number of vertex copies depends on how many domain vertices the two faces share, which is determined by: i) if the seam goes through the shared edge, and ii) if it ends in a shared vertex. The two triangle Jacobians are

$$J_1(X) = X' G'_1 \in \mathbb{R}^{2 \times 2}$$

$$J_2(X) = R'_m X' G'_2 R'_k \in \mathbb{R}^{2 \times 2},$$

where  $G_1, G_2 \in \mathbb{R}^{2 \times n}$  are discrete differential operators (Appendix A).  $J_2$  is transformed to the local coordinate frame of  $J_1$  via a connection:  $R_k$  is a rotation matrix that aligns the shared edge in the frames, and  $R_m$  is a rotation matrix based on the matching (period jump) of the edge [1].

Define the functions:

$$\alpha(X) = J_1(X) - J_2(X) \in \mathbb{R}^{2 \times 2}$$

$$\eta(X) = \|\alpha(X)\|_F^2 \in \mathbb{R}$$

$$\begin{aligned}\beta(J) &= |J| \in \mathbb{R} \\ \mu(J) &= \beta(J)^{-1} \in \mathbb{R} \\ \psi_i(X) &= \eta(X) \cdot \mu(J_i(X)) \in \mathbb{R}.\end{aligned}$$

The energy is

$$\psi(X) = \psi_1(X) + \psi_2(X) \in \mathbb{R}.$$

### Appendix C.1. First Derivative

$$\mathbf{D} J_1(X) = (G_1 \otimes I_2) K_{n2} \in \mathbb{R}^{4 \times 2n}$$

$$\mathbf{D} J_2(X) = (R_\kappa G_2 \otimes I_2) \cdot (I_n \otimes R'_m) \cdot K_{n2} \in \mathbb{R}^{4 \times 2n}$$

$$\mathbf{D} \alpha(X) = [(G_1 \otimes I_2) - (R_\kappa G_2 \otimes I_2) \cdot (I_n \otimes R'_m)] K_{n2} \in \mathbb{R}^{4 \times 2n}$$

$$\mathbf{D} \eta(X) = 2(\text{vec } \alpha)' \cdot \mathbf{D} \alpha(X) \in \mathbb{R}^{1 \times 2n}$$

$$\mathbf{D} \beta(J) = (\text{vec } |J| (J^{-1})')' \in \mathbb{R}^{1 \times 4}$$

$$\mathbf{D} \mu(\beta) = -\beta^{-2} \in \mathbb{R}$$

$$\mathbf{D} \mu(J) = (\mathbf{D} \mu(\beta)) \cdot \mathbf{D} \beta(J) \in \mathbb{R}^{1 \times 4}$$

$$\mathbf{D}_X \mu(J) = (\mathbf{D} \mu(J)) \cdot \mathbf{D} J(X) \in \mathbb{R}^{1 \times 2n},$$

where  $\mathbf{D} \mu(\beta)$ ,  $\mathbf{D} \mu(J)$ , and  $\mathbf{D}_X \mu(J)$  are the Jacobians of  $\mu$  with respect to  $\beta$ ,  $J$ , and  $X$ , respectively.

$$\mathbf{D} \psi_i(X) = (\mathbf{D} \eta(X)) \mu(J_i) + \eta(X) \mathbf{D}_X \mu(J_i) \in \mathbb{R}^{1 \times 2n}.$$

The Jacobian is:

$$\mathbf{D} \psi(X) = \mathbf{D} \psi_1(X) + \mathbf{D} \psi_2(X) \in \mathbb{R}^{1 \times 2n}.$$

### Appendix C.2. Second Derivative

$$\mathbf{H} \eta(X) = (\mathbf{D} \alpha)' \cdot 2I_4 \cdot \mathbf{D} \alpha \in \mathbb{R}^{2n \times 2n}$$

$$\mathbf{H} \beta(J) = |J| K_2 (J^{-1} \otimes I_2)' \cdot ((\text{vec } I_2)(\text{vec } I_2)' - I_4) \cdot (I_2 \otimes J^{-1}) \in \mathbb{R}^{4 \times 4}$$

$$\mathbf{H} \mu(J) = (\mathbf{D} \beta(J))' \cdot 2\beta^{-3} \mathbf{D} \beta(J) + (\mathbf{D} \mu(\beta)) \cdot \mathbf{H} \beta(J) \in \mathbb{R}^{4 \times 4}$$

$$\mathbf{H}_X \mu(J) = (\mathbf{D} J)' \cdot (\mathbf{H} \mu(J)) \cdot \mathbf{D} J \in \mathbb{R}^{2n \times 2n},$$

where  $\mathbf{H} \mu(J)$  and  $\mathbf{H}_X \mu(J)$  are the Hessians of  $\mu$  with respect to  $J$  and  $X$ , respectively.

$$\mathbf{H} \psi_i(X) = \text{sym} [(\mathbf{H} \eta(X)) \mu(J_i) + 2(\mathbf{D} \eta(X))' \cdot \mathbf{D}_X \mu(J_i) + \eta(X) \mathbf{H}_X \mu(J_i)] \in \mathbb{R}^{2n \times 2n}.$$

The Hessian is

$$\mathbf{H} \psi(X) = \mathbf{H} \psi_1(X) + \mathbf{H} \psi_2(X) \in \mathbb{R}^{2n \times 2n}.$$

## Appendix D. Frame Alignment Energy Differentiation

### Appendix D.1. First Derivative

$$\begin{aligned}\mathbf{D} J^{-1}(X) &= -\left(J^{-1}(X)\right)' \otimes J^{-1}(X) \cdot \mathbf{D} J(X) \in \mathbb{R}^{4 \times 6} \\ \mathbf{D} \xi(J) &= \text{vec}(J - A)' \in \mathbb{R}^{1 \times 4} \\ \mathbf{D} \xi\left(J^{-1}\right) &= \text{vec}\left(J^{-1} - A^{-1}\right)' \in \mathbb{R}^{1 \times 4}.\end{aligned}$$

The Jacobian of  $\xi$  with respect to  $X$  is

$$\mathbf{D} \xi(X) = \mathbf{D} \xi(J) \cdot \mathbf{D} J(X) + \mathbf{D} \xi\left(J^{-1}\right) \cdot \mathbf{D} J^{-1}(X) \in \mathbb{R}^{1 \times 6}. \quad (\text{D.1})$$

### Appendix D.2. Second Derivative

$$\begin{aligned}\mathbf{H} c_{ij;2}\left(J^{-1}\right) &= 2K'_2\left(J^{-1}E_{ij;2}J^{-1} \otimes I_2\right)' \cdot \left(I_2 \otimes J^{-1}\right) \in \mathbb{R}^{4 \times 4} \\ \mathbf{H} J^{-1}(J) &= \text{sym} \begin{bmatrix} \mathbf{H} c_{11;2}\left(J^{-1}\right) \\ \mathbf{H} c_{21;2}\left(J^{-1}\right) \\ \mathbf{H} c_{12;2}\left(J^{-1}\right) \\ \mathbf{H} c_{22;2}\left(J^{-1}\right) \end{bmatrix}_{16 \times 4} \\ \mathbf{H} J^{-1}(X) &= (I_4 \otimes \mathbf{D} J(X))' \cdot \left(\mathbf{H} J^{-1}(J)\right) \cdot \mathbf{D} J(X) \in \mathbb{R}^{24 \times 6}\end{aligned}$$

The Hessian of  $\xi$  with respect to  $X$  is

$$\mathbf{H} \xi(X) = (\mathbf{D} J(X))' \cdot \mathbf{D} J(X) + \left(\mathbf{D} J^{-1}(X)\right)' \cdot \mathbf{D} J^{-1}(X) + \left(\mathbf{D} \xi\left(J^{-1}\right) \otimes I_6\right) \cdot \mathbf{H} J^{-1}(X) \in \mathbb{R}^{6 \times 6}. \quad (\text{D.2})$$

## References

- [1] D. Bommes, H. Zimmer, L. Kobbelt, Mixed-integer quadrangulation, ACM Trans. Graph. 28 (3) (2009) 77:1–77:10.
- [2] A. Myles, D. Zorin, Global parametrization by incremental flattening, TOG 31 (4) (2012) 109.
- [3] Z. Levi, Direct seamless parametrization, ACM Transactions on Graphics 40 (1) (2021) 1–14.
- [4] M. Campen, D. Bommes, L. Kobbelt, Quantized global parametrization, ACM Trans. Graph. 34 (6) (2015) 192:1–192:12.
- [5] P. M. Knupp, C. Ernst, D. C. Thompson, C. Stimpson, P. P. Pebay, The verdict geometric quality library., Tech. rep., Sandia National Laboratories (2007).
- [6] D. Panozzo, E. Puppo, M. Tarini, O. Sorkine-Hornung, Frame fields: anisotropic and non-orthogonal cross fields, ACM Trans. Graph. 33 (4) (2014) 134:1–134:11.
- [7] A. O. Sagedan-Furnas, A. Chern, M. Ben-Chen, A. Vaxman, Chebyshev nets from commuting polyvector fields, ACM Transactions on Graphics 38 (6) (2019) 1–16.
- [8] W. Jakob, M. Tarini, D. Panozzo, O. Sorkine-Hornung, Instant field-aligned meshes, ACM Trans. Graph. 34 (6) (2015) 189–1.
- [9] J. Huang, Y. Zhou, M. Niessner, J. R. Shewchuk, L. J. Guibas, Quadriflow: A scalable and robust method for quadrangulation, in: Computer Graphics Forum, Vol. 37, 2018, pp. 147–160.
- [10] M. Campen, M. Ibing, H.-C. Ebke, D. Zorin, L. Kobbelt, Scale-invariant directional alignment of surface parametrizations, in: Computer Graphics Forum, Vol. 35, 2016, pp. 1–10.
- [11] J. R. Shewchuk, What is a good linear finite element? - interpolation, conditioning, anisotropy, and quality measures, Tech. rep., In Proc. of the 11th International Meshing Roundtable (2002).
- [12] X. Gao, J. Huang, K. Xu, Z. Pan, Z. Deng, G. Chen, Evaluating hex-mesh quality metrics via correlation analysis, in: Computer Graphics Forum, Vol. 36, 2017, pp. 105–116.
- [13] P. M. Knupp, Algebraic mesh quality metrics, SIAM journal on scientific computing 23 (1) (2001) 193–218.
- [14] P. M. Knupp, Algebraic mesh quality metrics for unstructured initial meshes, Finite Elements in Analysis and Design 39 (3) (2003) 217–241.
- [15] J. Robinson, Cre method of element testing and the jacobian shape parameters, Engineering Computations.
- [16] K. Crane, Conformal geometry of simplicial surfaces, An Excursion Through Discrete Differential Geometry: AMS Short Course 76 (2020) 59.
- [17] Y. Lipman, Bounded distortion mapping spaces for triangular meshes, ACM Transactions on Graphics (TOG) 31 (4) (2012) 108.
- [18] D. Bommes, B. Lévy, N. Pietroni, E. Puppo, C. Silva, M. Tarini, D. Zorin, Quad-mesh generation and processing: A survey, CGF 32 (6) (2013) 51–76.

- [19] M. Marinov, L. Kobbelt, A robust two-step procedure for quad-dominant remeshing, in: Computer Graphics Forum, Vol. 25, 2006, pp. 537–546.
- [20] X. Liang, M. S. Ebeida, Y. Zhang, Guaranteed-quality all-quadrilateral mesh generation with feature preservation, Computer Methods in Applied Mechanics and Engineering 199 (29–32) (2010) 2072–2083.
- [21] F. B. Atalay, S. Ramaswami, D. Xu, Quadrilateral meshes with provable angle bounds, Engineering with Computers 28 (1) (2012) 31–56.
- [22] A. A. Rushdi, S. A. Mitchell, A. H. Mahmoud, C. C. Bajaj, M. S. Ebeida, All-quad meshing without cleanup, Computer-Aided Design 85 (2017) 83–98.
- [23] P. M. Knupp, Achieving finite element mesh quality via optimization of the jacobian matrix norm and associated quantities. part i—a framework for surface mesh optimization, International Journal for Numerical Methods in Engineering 48 (3) (2000) 401–420.
- [24] Y. Zhang, C. Bajaj, G. Xu, Surface smoothing and quality improvement of quadrilateral/hexahedral meshes with geometric flow, Communications in Numerical Methods in Engineering 25 (1) (2009) 1–18.
- [25] W. Schroeder, K. Martin, B. Lorensen, The visualization toolkit an object-oriented approach to 3d graphics fourth edition.
- [26] A. Johnen, C. Geuzaine, T. Toulorge, J.-F. Remacle, Efficient computation of the minimum of shape quality measures on curvilinear finite elements, Procedia engineering 163 (2016) 328–339.
- [27] X. Fang, H. Bao, Y. Tong, M. Desbrun, J. Huang, Quadrangulation through morse-parameterization hybridization, ACM Trans. Graph. 37 (4) (2018) 92:1–92:15.
- [28] J. Daniels, C. T. Silva, J. Shepherd, E. Cohen, Quadrilateral mesh simplification, ACM transactions on graphics 27 (5) (2008) 1–9.
- [29] S. A. Canann, J. R. Tristano, M. L. Staten, et al., An approach to combined laplacian and optimization-based smoothing for triangular, quadrilateral, and quad-dominant meshes., in: IMR, 1998, pp. 479–494.
- [30] R. V. Garimella, M. J. Shashkov, P. M. Knupp, Triangular and quadrilateral surface mesh quality optimization using local parametrization, Computer Methods in Applied Mechanics and Engineering 193 (9–11) (2004) 913–928.
- [31] S. Bouaziz, M. Deuss, Y. Schwartzburg, T. Weise, M. Pauly, Shape-up: Shaping discrete geometry with projections, in: Computer Graphics Forum, Vol. 31, 2012, pp. 1657–1667.
- [32] K. Xu, X. Gao, G. Chen, Hexahedral mesh quality improvement via edge-angle optimization, Computers & Graphics 70 (2018) 17–27.
- [33] N. Pietroni, C. Dumery, R. Falque, M. Liu, T. Vidal-Calleja, O. Sorkine-Hornung, Computational pattern making from 3D garment models, ACM Transactions on Graphics 41 (4) (2022) 157:1–14.
- [34] J. McCartney, B. Hinds, B. Seow, D. Gong, An energy based model for the flattening of woven fabrics, Journal of Materials Processing Technology 107 (1–3) (2000) 312–318.
- [35] J. McCartney, B. Hinds, K. Chong, Pattern flattening for orthotropic materials, Computer-Aided Design 37 (6) (2005) 631–644.
- [36] D. Kovacs, A. Myles, D. Zorin, Anisotropic quadrangulation, Computer Aided Geometric Design 28 (8) (2011) 449–462.
- [37] N. Aigerman, R. Poranne, Y. Lipman, Lifted bijections for low distortion surface mappings, TOG 33 (4) (2014) 69.
- [38] O. Weber, A. Myles, D. Zorin, Computing extremal quasiconformal maps, CGF 31 (5) (2012) 1679–1689.
- [39] J. Smith, S. Schaefer, Bijective parameterization with free boundaries, ACM Trans. Graph. 34 (4) (2015) 70:1–70:9.
- [40] A. Shtengel, R. Poranne, O. Sorkine-Hornung, S. Z. Kovalevsky, Y. Lipman, Geometric optimization via composite majorization, ACM Trans. Graph. 36 (4).
- [41] M. Rabinovich, R. Poranne, D. Panizzo, O. Sorkine-Hornung, Scalable locally injective mappings, ACM Trans. Graph. 36 (4).
- [42] K. Hormann, G. Greiner, Mips: An efficient global parametrization method, Curve and Surface Design: Saint-Malo 99 (2000) 153–162.
- [43] B. Smith, F. D. Goes, T. Kim, Analytic eigensystems for isotropic distortion energies, ACM Transactions on Graphics (TOG) 38 (1) (2019) 1–15.
- [44] A. Jacobson, E. Tosun, O. Sorkine, D. Zorin, Mixed finite elements for variational surface modeling, in: Computer Graphics Forum, Vol. 29, 2010, pp. 1565–1574.
- [45] K. Crane, U. Pinkall, P. Schröder, Robust fairing via conformal curvature flow, TOG 32 (4) (2013) 1–10.
- [46] J. Teran, E. Sifakis, G. Irving, R. Fedkiw, Robust quasistatic finite elements and flesh simulation, in: Proceedings of SCA, 2005, pp. 181–190.
- [47] Z. Levi, D. Zorin, Strict minimizers for geometric optimization, ACM Transactions on Graphics (TOG) 33 (6) (2014) 185.
- [48] J. R. Magnus, H. Neudecker, Matrix differential calculus with applications in statistics and econometrics, John Wiley & Sons, 2019.
- [49] U. Pinkall, K. Polthier, Computing discrete minimal surfaces and their conjugates, Experimental mathematics 2 (1) (1993) 15–36.