# Fields on Symmetric Surfaces

Daniele Panozzo[1,2,4] Yaron Lipman[3] Enrico Puppo[1] Denis Zorin[4]
[1]University of Genova [2]ETH Zurich [3]Weizmann Institute of Science [4]New York University

## Abstract

Direction fields, line fields and cross fields are used in a variety of computer graphics applications ranging from non-photorealistic rendering to remeshing. In many cases, it is desirable that fields adhere to symmetry, which is predominant in natural as well as man-made shapes. We present an algorithm for designing smooth N-symmetry fields on surfaces respecting *generalized symmetries* of the shape, while maintaining alignment with local features. Our formulation for constructing symmetry fields is based on global symmetries, which are given as input to the algorithm, with no isometry assumptions. We explore in detail the properties of generalized symmetries (reflections in particular), and we also develop an algorithm for the robust computation of such symmetry maps, based on a small number of correspondences, for surfaces of genus zero.

**CR Categories:** I.3.5 [Computer graphics]: Computations geometry and object modeling—Curve, surface, solid and object repres.

**Keywords:** symmetry, n-rosy field, quad mesh

**Links:** DL PDF

## 1 Introduction

Many geometry processing applications require the construction of $N$-symmetry (a.k.a. $N$-RoSy) fields on surfaces, i.e., fields that associate to every point a set of $N$ unary vectors forming equal angles between radially consecutive directions. For example, a line (2-symmetry) field can be used to guide texture placement or texture synthesis, as well as for anisotropic smoothing or text placement; a cross (4-symmetry) field is useful for constructing quadrangulations, for anisotropic remeshing, and for supporting non-photorealistic rendering; 6-symmetry fields have been used very recently to support hexagonal parametrization [Nieser et al. 2012].

In most cases, it is desirable for the fields to respect the symmetries of surfaces: meshes respecting symmetry are visually preferable and reduce deformation/animation artifacts; patterns and small-scale geometry (for example, fish scales or fur) are often required to follow the shape's symmetries. Figure 1 shows field-aligned parametrization using cross field constructed with our method, side by side with the mixed-integer quadrangulation (MIQ) field of Bommes *et. al.*[2009]. We highlight field singularities, as well as some integral lines to emphasize the symmetry aspect of the fields.

The goal of this work is to provide an algorithm for the construction of quasi-symmetric $N$-symmetry fields on surfaces with symmetry, which strikes a balance between three important properties of fields: symmetry, smoothness, and alignment with local geometry; we found this combination to be essential: while coarse global symmetry is highly desirable, alignment with local, possibly non-symmetric features is necessary as well, and smoothness must be always guaranteed. These fields can be used, e.g., to generate a quadrangulation; there is no guarantee that the resulting quadrangulation is fully symmetric, though.

To construct symmetric fields, we need to choose a class of surface symmetries (i.e., mappings of the surface to itself identifying symmetric points) that the fields can adhere to, and methods to construct such mappings. Most works on symmetry detection consider isometric maps (either intrinsic or extrinsic) as the principal model for symmetries, and make the assumption that the relevant symmetries are close to isometric. We formulate our algorithm for a broader class of *generalized symmetries* based on surface diffeomorphisms, without making any assumption about isometry.

There are three-fold advantages to this approach: (1) our method for symmetric field construction is less dependent on specific assumptions about the symmetry maps; (2) we can handle significant local deviations from isometry gracefully; and (3) in the case of genus zero surfaces, the concept of generalized symmetries and their properties lead to a robust and efficient algorithm for computing symmetry maps from a small number of correspondences. While our symmetric field construction algorithm can use symmetry maps produced by different algorithms, we demonstrate that the new algorithm yields substantially better quality.

To summarize, the main contributions of our work include

1. the introduction of generalized symmetries, with focus on generalized reflections, and invariant $N$-symmetry fields;

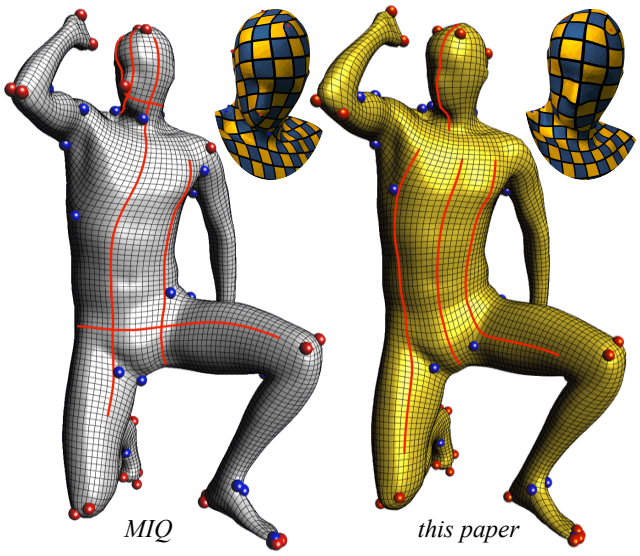


**Figure 1:** *Field-aligned parametrization of the knelt human model using the symmetry field construction method developed in this paper, and using the MIQ technique of Bommes* et. al.*[2009]. Red/blue bullets represent field singularities with positive/negative index. Red lines trace flows of the cross field.*

2. an algorithm that takes in an input symmetry map and builds a quasi-symmetric $N$-symmetry fields, which maintains alignment to local (possibly asymmetric) features - the single tunable parameter is used to adjust the relative importance of symmetry vs smoothness in the output (see Figure 2);

3. an algorithm for computing generalized intrinsic reflection maps of surfaces of genus zero, providing robust input for the symmetric field computation algorithm.

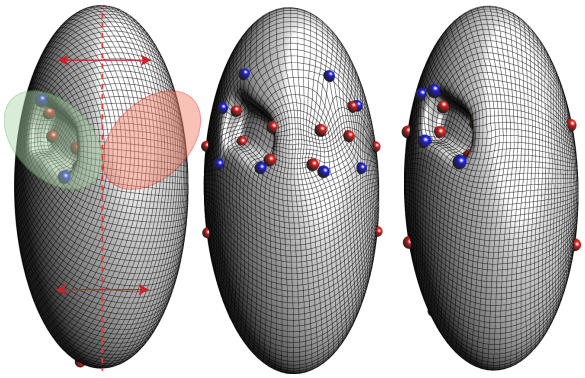

**Figure 2:** *Global symmetry vs local alignment on a synthetic model: (left) non-symmetric field obtained without symmetry detection; note the global bias introduced by local features; (center) direct field symmetrization transfers singularities also to the smooth side; (right) local alignment allows to combine global symmetry with local feature alignment.*

## 2 Related work

**N-symmetry fields.** A variety of methods were proposed for cross field and more generally $N$-symmetry field construction. A number of methods rely on manually placed singularities and other user-specified information [Ray et al. 2008; Crane et al. 2010; Lai et al. 2010]. As for shapes even of moderate complexity defining all singularities especially in a symmetric way is difficult, our method computes singularity positions automatically, based on the Mixed-Integer (MIQ) algorithm of [Bommes et al. 2009]. In this method, extending [Ray et al. 2008], fields are represented by per-triangle angles in fixed frames, and *matchings* on edges, indicating the additional $k\pi/N$ rotation of the field to transition between adjacent faces. MIQ [Bommes et al. 2009] uses a greedy strategy to make the field as smooth as possible, with both angles and integers $k$ as variables. Other methods include nonlinear optimization [Hertzmann and Zorin 2000; Ray et al. 2006; Ray et al. 2009] also using angles to represent fields, and a linear optimization of a tensor field representation [Palacios and Zhang 2007]. As far as we know, no construction takes symmetry of the domain into account.

Line fields, cross fields and 6-symmetry fields have been used to compute mesh parametrization and surface remeshing. *Field-alignment techniques* [Ray et al. 2006; Kälberer et al. 2007; Bommes et al. 2009; Nieser et al. 2012] adapt a parameterization to a shape by fitting the parametrization gradient to smoothed principal curvature directions, or more generally, to a smooth field capturing surface features. The topological structure of the field (singularities and separating lines) indirectly determines how fine the domain mesh can be. Some recent results address domain simplification by either reducing the number of singularities [Peng et al. 2011] or disentangling the graph of separating lines [Bommes et al. 2011; Tarini et al. 2011].

**Symmetry detection methods.** Several techniques based on voting have been proposed in the literature to detect either global or partial extrinsic symmetries. In [Podolak et al. 2006; Cailliere et al. 2008], symmetry planes are detected, while symmetric patches are found in [Mitra et al. 2006]. PIRS [Xu et al. 2009] extends the voting approach to find stationary lines of partial intrinsic symmetries. However, PIRS does not provide a dense map of correspondences, and the detection of the stationary line for approximate (non-isometric) symmetries is not stable enough to support our algorithm.

Other techniques reduce intrinsic symmetry to extrinsic by "straightening" objects through deformation. [Mitra et al. 2007] proposed a fully automatic method, which works for symmetric objects with multiple extrinsic symmetries, based on [Mitra et al. 2006]. A similar approach is proposed in [Kazhdan et al. 2009; Ghosh et al. 2010], requiring manual input for the starting phase. These approaches do not provide a dense map of correspondences.

In [Golovinskiy et al. 2009; Podolak et al. 2007] some of the above methods are combined with mesh simplification techniques to produce symmetric triangle meshes.

A few other works address intrinsic symmetry through embedding methods that either reduce intrinsic to extrinsic symmetry [Ovsjanikov et al. 2008], or factor out symmetry by mapping sets of symmetric points to a single point in an embedding space [Ovsjanikov et al. 2010; Lipman et al. 2010]. These techniques are fully automatic; they may provide a dense (in a vertex-to-vertex sense) mapping of symmetries; and they may be easily generalized to find the stationary line. However, they are not robust to surface deformations that break isometry.

Embedding techniques, as well as voting techniques, do not exploit spatial coherence: point-to-point symmetry is estimated without taking into account what happens at nearby points. For this reason, they may be prone to errors such as false matchings and discontinuous mapping, which may severely hinder the application of extracted maps for our purposes.

Most recent approaches exploit spatial coherence by using surface parametrization: [Kim et al. 2010; Kim et al. 2011] present fully automatic methods that first find sparse matchings of symmetric points, and then apply a Möbius transform that realizes extrinsic symmetry in parameter space, assuming a topologically restricted, conforming parametrization. These methods can provide continuous mappings of symmetries, but they are prone to severe errors in the presence of false matchings during the coarse phase.

To the best of our knowledge, approximate intrinsic symmetries in the non-isometric case have been addressed only in [Raviv et al. 2007; Raviv et al. 2010] by using an alternative definition to the one we present in Section 3. However, their definition (even in the continuous case) does not imply a smooth diffeomorphism: the derived algorithms are combinatorial in nature and they are meant more to assess the degree of non-isometry than to detect an explicit map of symmetry.

Given a set of few symmetric landmark pairs, one can think of using inter-surface mapping methods for building the symmetry map. A common approach is to find a common parametrization domain for the two surfaces [Praun et al. 2001]. Later work [Kraevoy and Sheffer 2004; Schreiner et al. 2004] developed automatic algorithms to find a suitable base parameter meshes. However, these methods are general and do not exploit the fact that the final map should be a *symmetry*. For example, in case of bilateral reflective symmetry these methods will not force the existence of a stationary closed curve, as we require for our field construction.

## 3 Symmetric fields

Symmetries on a surface are usually defined as isometric automorphisms: *extrinsic* symmetries preserve Euclidean distance, while

*intrinsic* symmetries preserve geodesic distance. The class of intrinsic symmetries trivially includes the class of extrinsic ones.

In practice, few surfaces have perfectly isometric symmetries, and deviations, sometimes quite large, need to be allowed.

To be able to handle such maps, we regard *any* smooth automorphism of a surface as a symmetry, and we focus on topological properties, as it is common in mathematical study of symmetries of surfaces (cf. [Farb and Margalit 2011]).

We consider $N$-symmetry fields defined on 2-manifolds, as defined in [Ray et al. 2008], and we study the properties that one such field must have to comply with a given symmetry on its domain. In Section 6, we discuss the generalization to *sets of symmetries*.

### 3.1 Generalized reflections

We focus on *reflections*, which account for most global symmetries observed in real objects. A reflection $g$ and identity form a finite group of transformations, and each point has an orbit with respect to this group consisting of two points. In the extrinsic isometric case, all global symmetries for compact objects can be composed of reflections, as any 3D rotation can be decomposed into two reflections. While objects with rotational but no reflectional symmetries do exist, they are relatively rare.

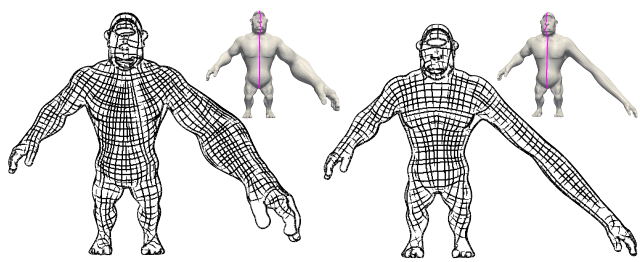

**Figure 3:** *Two models with non-isometric symmetry: the stationary line is depicted in magenta; our algorithm computes a field that respects these generalized symmetries. Fields rendered with hatching from [Palacios and Zhang 2011].*

A diffeomorphism $g : M \to M$ is a smooth map with a smooth inverse from the surface to itself. $M(g)$ denotes the *stationary set* of $g$, i.e., the set of points $p$ of $M$ for which $g(p) = p$. For a point $p$ on $M$, $T_pM$ is the tangent plane at $p$. The differential of $g$ at $p$, $Dg_p : T_pM \to T_{g(p)}M$ maps tangent vectors at $p$ to tangent vectors at $g(p)$.

**Generalized reflections.** We adopt the following definition of a *generalized reflection* (cf. [Koszul 1965]):
**Definition 1.** *A* reflection *on $M$ is a diffeomorphism $g$, such that $g(g(p)) = p$ for all $p \in M$, and the set of non-stationary points of $g$ is nonempty (hence $g$ is not the identity) and disconnected.*

Stationary points of symmetry mappings $g$ play a particularly important role in the symmetric field construction of Section 4. For a generalized reflection $g$, it turns out that the local behavior near stationary points is similar (although not identical) to the behavior of isometric reflections.
**Lemma 1.** *Let $g$ be a reflection. If $p$ is a stationary point of $g$, then:*

1. *the differential $Dg_p$ has a stationary direction $s_p$;*
2. *for a choice of an orthonormal coordinate system on $T_p$ the differential $Dg_p$ has the form*

$$\begin{bmatrix} 1 & c \\ 0 & -1 \end{bmatrix}.$$

3. *[Montgomery and Zippin 1955] There is a neighborhood $U(p)$, and a choice of smooth coordinates $h : U \to \mathbb{R}^2$ system on $U$ such that $g$ in these coordinates is a linear transformation $A_g$, i.e.*

$$g = h^{-1} \circ A_g \circ h \tag{1}$$

We prove this Lemma in the supplementary material.

In the proximity of the stationary line, $Dg$ behaves as a linear reflection combined with a shear, and the value of factor $c$ determines the amount of shear. If a map is conformal it follows that $c = 0$ and $g$ is isometric at stationary points.

*Global* properties of generalized reflections are also similar to the familiar reflections about a symmetry plane of an object [Koszul 1965]. More specifically, the following proposition holds.
**Proposition 2.** *If $g$ is a reflection on $M$, then:*

1. *$g$ is orientation-reversing;*
2. *the stationary set of $g$ is a set of closed smooth curves on $M$ (it generalizes intersection with the symmetry plane);*
3. *$M' = M \setminus M(g)$ consists of two connected components $M_1$ and $M_2$*
4. *$g$ maps $M_1$ and $M_2$ to each other.*

An important consequence of Proposition 2 is the following.
**Corollary 3.** *For surfaces of genus zero* (1) *holds globally, i.e. for a generalized reflection $g$ there is a diffeomorphism onto the extended complex plane*[1] *$h : M \to \widehat{\mathbb{C}}$, such that $g = h^{-1} \circ A \circ h$, where $A$ is a reflection, $A(z) = \overline{z}$.*

The proof of the proposition and the corollary can be found in the supplemental material. We construct this type of global parametrization for genus zero surfaces in Section 5.

### 3.2 $N$-symmetry fields

A *$N$-symmetry field* $v$ on the surface is an assignment to every point $p \in M$ (excluding a set isolated singularities) of $N$ unit vectors $v_1, \ldots, v_N$ lying on the tangent plane $T_pM$ and forming equal angles of $2\pi/N$ between adjacent vectors. The most common and useful examples are direction fields, line fields, cross fields (i.e., 1-symmetry, 2-symmetry and 4-symmetry fields, respectively) and 6-symmetry fields. We primarily focus on cross fields, but the algorithm described in Section 4 applies to any value of $N$, and would be easily adapted to non-unit fields.

**Transport of $N$-symmetry fields.** To define $N$-symmetry fields that respect a symmetry map $g$, we need a way to compare the values of the field at symmetric points. In order to do this, we must be able to *transport* the symmetry field at a given point $p$ to the tangent plane at $g(p)$. The differential $Dg$ defines a natural map $T_pM \to T_{g(p)}M$ for vector fields. If $Dg$ is orthogonal, then it can be trivially extended to transport any $N$-symmetry field: the $N$ vectors $Dg\, v_i(p)$ form a $N$-symmetry value (a set of $N$ unit-length vectors separated by equal angles). In a more general case of $N$-symmetry fields, $Dg$ applied to the component vectors of the field does not yield a new $N$-symmetry value unless $Dg$ is an isometry. Broadening the class of fields we consider is possible, but undesirable, as many target algorithms expect orthogonal directions. Instead, we define an orthogonal transport $T_pM \to T_{g(p)}M$ associated with $g$, as the closest orthogonal transform $R^g$ to $Dg$. As $g$ is orientation-reversing, this transform is a reflection. See Figure 4 for illustration.

[1] The extended complex plane is the complex plane $\mathbb{C} = \{x + \mathrm{i}y \mid x, y \in \mathbb{R}\}$ with infinity added to it, that is $\widehat{\mathbb{C}} = \mathbb{C} \cup \{\infty\}$.

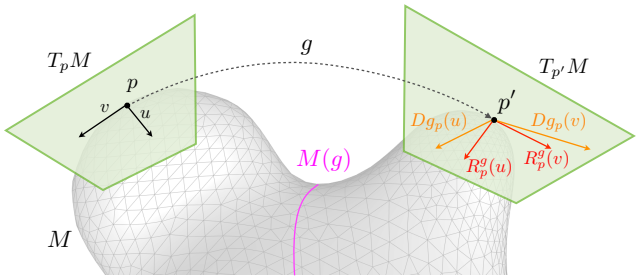


**Figure 4:** *Summary of notation for surface $M$ with non-isometric reflection $g$. Orthonormal vectors $u$ and $v$ are transported to non-orthonormal orange vectors by differential $Dg$ and to orthonormal red vectors by its closest orthogonal transform $R^g$.*

We map the $N$-symmetry field $v$ using the closest orthogonal transform $R^g$ to $Dg$, which can be obtained from the unique polar decomposition

$$Dg = R^g S^g, \tag{2}$$

where $R^g$ is orthonormal and $S^g$ is symmetric positive definite[2]. Note that, if $g$ is an isometry, we trivially have $Dg = R^g$.

**Lemma 4.** *$R^g(p) : T_pM \to T_{g(p)}M$ continuously depends on $Dg$, and if $g$ is a reflection, then for every $p \in M$, and for any $N$-symmetry field $v$, we have $v(p) = R^g(g(p))R^g(p)v(p)$.*

The Lemma is proved in the supplemental material.

We can now define the symmetric $N$-symmetry fields:

**Definition 2.** *A $N$-symmetry field $v$ is* symmetric with respect to a symmetry map *$g$ if either $p$ and $g(p)$ are both singularities of $v$, or*

$$R^g_p(p)v(p) = v(g(p)), \tag{3}$$

*where $R^g_p$ is the closest orthonormal transform to the differential $Dg_p$ at a point $p$, as defined above.*

**$N$-symmetry fields on stationary lines.** Symmetric $N$-symmetry fields are highly constrained at stationary points. The next corollary follows from Lemma 1 and definition of $R^g$ by observing that $Dg$ at stationary points is a combination of linear reflection and shear and so $R^g$ at that point is just a linear reflection:

**Corollary 5.** *If $p$ is a stationary point of $g$, then $R^g : T_pM \to T_pM$ is a reflection defined by a linear transform.*

By definition, if $v$ is a symmetric $N$-symmetry field, and $p$ is a stationary point, then we must have $R^g v(p) = v(p)$. The above corollary also indicates that $R^g$ is a linear reflection. This imposes stringent requirements on field $v$.

**Proposition 6.** *If $v$ is symmetric with respect to a symmetry transform $g$ and $p$ is a stationary point of $g$, then one of the following holds: (1) $v$ has a singularity at $p$, (2) one of the directions of $v$ is the stationary direction $s_p$ of $R^g_p$; (3) one of the bisectors of angles formed by consecutive vectors of $v$ is aligned with $s_p$.*

The proposition is proved in the supplemental material.

The proposition implies that only two possible orientations exist at each point of the stationary line for a symmetric field $v$. Moreover, by continuity of $v$, the same choice must hold at all points along a connected component of the stationary line, unless it contains a singularity. We will use this fact as a basis of our algorithm for computing a symmetric field. Figure 5 shows the two possible choices for cross field at stationary curve.

[2]These decompositions can be computed, e.g., either via SVD: $D_g = U\Sigma V^T$, hence $R^g = UV^T$ and $S^g = V\Sigma V^T$; or through an explicit formula, $R^g = B/\sqrt{|det(B)|}$, $B = (1/2)(Dg + Dg^T - Tr(Dg)I)$, with $Dg$ expressed as a $2 \times 2$ matrix in an orthonormal basis.

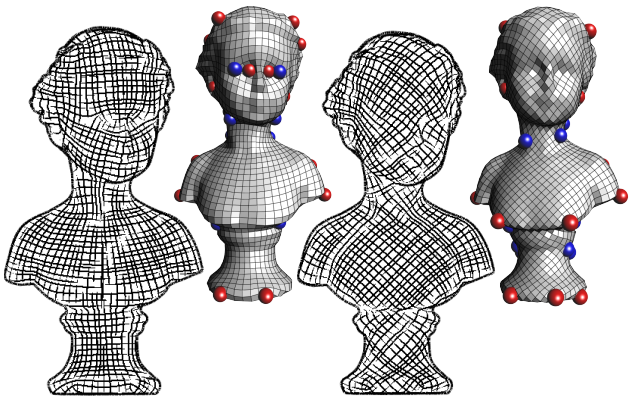

**Figure 5:** *A field can have just two possible orientations at the stationary line: results for a cross field either aligned with the stationary line, or rotated by $\pi/4$. Field singularities are depicted by blue/red bullets. Field rendered with [Palacios and Zhang 2011] and MIQ remeshing.*

## 4 Field symmetrization

Given a mesh $M$, a generalized reflection $g$, and a set of orientation constraints, our algorithm computes an $N$-symmetry field $v$ on $M$ that is smooth, symmetric with respect to $g$, and aligned with the orientation constraints. The objectives of symmetry and smooth alignment to local shape features may be in conflict. Our algorithm provides a trade-off between the two, with a smoothness parameter chosen by the user. The foundation of our algorithm is the field optimization phase of the MIQ [Bommes et al. 2009].

**Algorithm outline.** We assume that for the symmetry map $g$ it is possible to: evaluate the stationary set $M(g)$ (realized as a set of segments on a subset of triangles); evaluate the field transport $R^g$ (as rigid linear transform between triangles $t, t'$); and evaluate the orbit $\mathcal{O}(p)$ at any point $p$ (for $p \in t$, we calculate $g(p) \in t'$, on a triangle $t'$). Computing stationary lines, transport maps and orbits for specific symmetry maps will be addressed in Section 5. Our algorithm is not specific to a single reflection; we make no assumptions about the size of the orbit, or the specific origin of the transport map. This allows us to apply it in the case of sets of symmetries (Section 6).

Our method consists of the following steps:

1. Set hard orientation constraints at selected feature points.

2. Set additional hard constraints at the stationary curve of $g$, $M(g)$ (excluding conflicts with constraints from 1), by fixing one of the two possible orientations of $v$ at stationary points (see Prop. 6), then extend field $v$ to the rest of $M$ by running the MIQ smoothing algorithm. We found that constraining the field on stationary lines is crucial for high-quality results.

3. Use field transport $R^g$ (Equation (2)) to symmetrize field $v$, by averaging over orbits (excluding orientation feature points and stationary points). Denote the output field of this stage $\bar{v}$.

4. Repeat (once) Step 2, to obtain the final field $v$ using $\bar{v}$ as soft constraint (this is where the parameter controlling smoothness vs. symmetry comes in).

Next, we describe each step of the algorithm in detail. At the end of this section, we briefly discuss an alternative approach and the reasons why this particular algorithm structure was chosen.

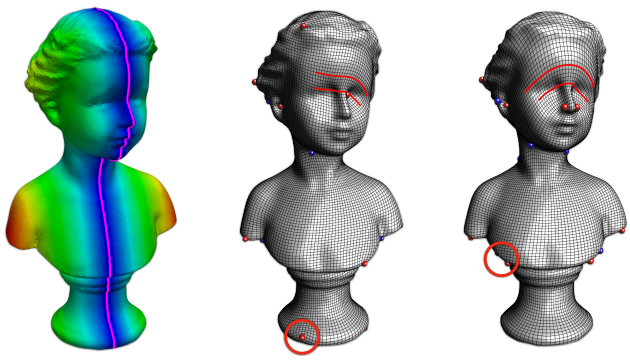

**Figure 6:** *Algorithm steps. From the left: input mesh with symmetry; after constrained field optimization algorithm follows stationary line but it is not symmetric everywhere; after symmetrization and final optimization field is fully symmetric. Red lines show field flow; circles highlight non-symmetric singularities that become symmetric after optimization.*

**Steps 1,2 - Constrained field optimization.** Similarly to [Ray et al. 2008; Bommes et al. 2009], a discrete field $v$ is represented at triangle $t_i$ by an angle $\theta_i$ with respect to a local frame. With each $e_{ij}$ separating triangles $t_i$ and $t_j$, we associate a constant angle $\kappa_{ij}$ and an integer variable $p_{ij}$ (*period jump* or *matching*). The angle $\kappa_{ij}$ is the rotation of the reference frame from $t_i$ to $t_j$. The period jump $p_{ij}$, determines the additional rotation $2\pi p_{ij}/N$ of the $N$-symmetry field between values on triangles $t_i$ and $t_j$. This rotation maps an $N$-symmetry value to itself, so it needs to be encoded separately (cf. [Ray et al. 2008])[3].

The smoothness energy of the MIQ algorithm is

$$E_{smooth} = \sum_{e_{ij} \in E} (\theta_i + \kappa_{ij} + \frac{2\pi}{N} p_{ij} - \theta_j)^2. \tag{4}$$

Since variables $\theta_i$ and $\theta_j$ are real, while the $p_{ij}$'s are integer, optimization of the field $v$ with respect to Energy (4) is a mixed-integer problem. This problem has a space of equivalent solutions: to make the minimizer unique, the value of $p_{ij}$ is fixed at a subset of edges. Feature constraints are given as a set of angles $\{\hat{\theta}_{i_1}, \ldots, \hat{\theta}_{i_k}\}$ on a subset of triangles which remain fixed during optimization; we add stationary lines constraints described below to this set. A greedy mixed-integer optimization algorithm is used for optimization. (see [Bommes et al. 2009] for details).

*Stationary set constraints.* For all $g \in G$, the stationary set $M(g)$ is given by a set of line segments $l_i$ on a subset of triangles of $M$. The direction of $l_i$ is the stationary direction of $R^g$ at $t_i$. We add a hard constraint by expressing the direction of $l_i$ as an angle $\hat{\theta}_i$ with respect to local coordinate frame at $t_i$. Alternatively, the angle $\hat{\theta}_i + \pi/N$ can be used, according to Prop. 6. This choice is left to the user and it must be consistent for all triangles intersecting a given connected component of $M(g)$. In our experiments, we always constrained $v$ to be aligned with the stationary line. Figure 5 shows results obtained by using the alternative direction.

Energy (4) is minimized by freezing all variables corresponding to hard constraints (see [Bommes et al. 2009] for details). The result of this phase already improves over the standard MIQ in terms of field symmetry in the proximity of the stationary line, while it does not warrant symmetry far from it (see Figure 6 center).

[3] We note that one can consider discrete principal connections instead of fields [Crane et al. 2010], replacing matchings with a more geometrically natural notion of holonomy angles; we prefer [Ray et al. 2008] formulation as it allows for more natural handling of constraints.

**Step 3 - Symmetrization by field transport.** Field $v$ is symmetrized by averaging it over orbits of symmetry. We set the convention that field values are attached to triangles' centroids $c_i \in t_i$.

*Fuzzy orbits.* Since discrete symmetry maps do not map triangles exactly to triangles, we use the notion of fuzzy orbits (similarly to [Lipman et al. 2010]) in order to define a symmetry-averaging operator. In particular, given a non-stationary triangle $t$, we define its fuzzy orbit $\mathcal{O}(t)$ as the union of all triangles in its 1-ring and the 1-ring of the triangle containing $g(c)$. We assign a weight $s_i(t)$ to every $t_i \in \mathcal{O}(t)$ inversely proportional to its distance from $\mathcal{O}(c) = \{c, g(c)\}$, namely we set

$$\widetilde{s}_i(t) = \Phi(\min_{c' \in \mathcal{O}(c)} \|c_i - c'\|),$$

where we picked $\Phi(r)$ to be a Gaussian with standard deviation equals the maximum of the triangles' 1-ring diameter. To define averaging, $\widetilde{s}_i(t)$ is normalized to have a unit sum:

$$s_i(t) = \widetilde{s}_i(t) / \sum_i \widetilde{s}_i(t).$$

*Averaging over orbits.* For every triangle $t$, we average the field values over the orbit $\mathcal{O}(t)$ using the weights $s_i(t)$ and the field transport $R^g$ as follows.

For every $t' \in \mathcal{O}(t)$ we transport the field value at $t'$ to $t$ using $R_{t',t}$. $R_{t',t}$ is defined as $R_{t',t} = (R^g_t)^{-1} R_{loc}$, where $R^g_t$ is the field transport from $t$ to the triangle $\widetilde{t}$ containing $g(c)$ ($c$ centroid of $t$), and $R_{loc}$ is the closest rotation in 3D taking $t'$ to $\widetilde{t}$. Following [Palacios and Zhang 2007], we observe that the $N$-symmetry field can be represented in a given frame by a vector $w = [\cos(N\theta), \sin(N\theta)]$, eliminating the $2\pi/N$ ambiguity. Then, instead of transporting the $N$-symmetry field vectors by $(R^g_t)^{-1}$ and then converting to the $N$-th power vectors, $w$ can be transported directly by $R^{-N}_{t',t}$. The symmetrized value of the vector field is defined explicitly by normalizing

$$w^{sym}(t) = \sum_i s_i(t) R^{-N}_{t_i,t} w(t_i).$$

**Step 4 - Optimization with soft symmetry constraints.** Field $\bar{v}$ obtained from the previous step will be mostly smooth, but smoothness can be broken in some parts of $M$, namely:

- in the proximity of hard constraints, since the field is fixed at such triangles, while it is possibly deviated by symmetrization at neighboring triangles;
- in the proximity of singularities and of their symmetric mates, since directions of the field are arbitrary at singularities, thus symmetrization may produce invalid results.

Discontinuities may also introduce many undesirable singularities in the field. Therefore, we smooth $\bar{v}$ further by minimizing a modified energy, using values $\bar{\theta}_i$ as soft constraints. Following [Bommes et al. 2009], we measure the *local roughness* of $\bar{v}$ at each triangle $t_i$ as the constant-weight discrete Laplacian of the field at $t_i$:

$$\delta_i = \min_{\bar{p}_{ij}} \sum_{t_j \in \mathcal{N}_i} (\bar{\theta}_i + \kappa_{ij} + \frac{2\pi}{N} \bar{p}_{ij} - \bar{\theta}_j)/3,$$

where $\mathcal{N}_i$ contains the three neighbors of $t_i$ and the $\bar{p}_{ij}$ can take values in $0, \ldots, N-1$. We note that this method is justified only for uniform triangulations, but we did not see a significant difference in practice, and using dual-mesh discrete Laplacian results in less stable behavior for badly shaped triangles. We define the symmetry constraint weight:

$$w(\delta_i) = \max(0, \delta_0 - \delta_i) / \max_i(\delta_0 - \delta),$$

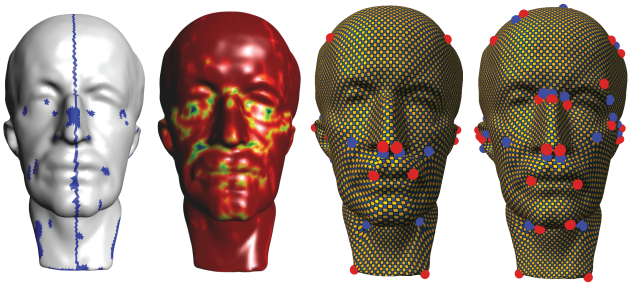

**Figure 7:** *Effect of the smoothness parameter. From left to right: curvature and symmetry constraints; smoothness of the symmetrized field; and final results with $\alpha = 0.1$ and $\alpha = 0.9$.*

which is zero if roughness exceeds a threshold $\delta_0$. We have used $\delta_0 = 10°$ in all our experiments. For a uniform triangulation, $\delta_0$ can be set automatically as the maximal rate of field rotation for the average triangle size. Finally, we define the modified energy as follows:

$$E_{symm} = (1-\alpha)E_{smooth} + \alpha \sum_{t_i \in M} w_i(\delta_i)(\theta_i - \bar{\theta}_i)^2, \qquad (5)$$

with the second term constraining the field values to be close to symmetric values. The final result is thus obtained by minimizing Energy (5) with the same hard constraints and the same mixed integer solver used in Step 2.

Parameter $\alpha$ is the main parameter of our method, and it is used explicitly to allow the user selecting either a smoother or a more symmetric result. In Figure 7, we report results obtained with different values of $\alpha$. Note that $\alpha = 0$ necessarily yields the same result that we obtain after Step 1; while, as $\alpha$ approaches value 1, field smoothing is progressively restricted just to areas where roughness exceeds threshold $\delta_0$.

**Alternative approaches.** A natural alternative method for field symmetrization is to formulate the field construction as a single energy minimization, with terms penalizing the difference between a the field at a point with the transported value of the field at the symmetric point. We have explored this approach in our initial version of the algorithm. Resulting fields did not show much improvement compared to MIQ fields even in the extrinsic case and with a manually placed symmetry plane, due to two main reasons: (a) in the generic case of an asymmetric mesh connectivity, it is hard to design a MI field optimization algorithm that works symmetrically; (b) it was difficult to provide a good-quality trade-off between *asymmetric* local feature alignment constraints, and global symmetry. Problem (a) in our algorithm is to great extent resolved by hard constraints on the symmetry line. Symmetrization followed by smooth interpolation in our algorithm resolves (b): we are able to produce a globally symmetric field with controllable smooth transition near feature constraints.

## 5 Symmetry detection algorithms

Our construction of symmetric $N$-symmetry fields requires a generalized reflection mapping $g$, as well as ways to evaluate its stationary set $M(g)$, and its transport $R^g$, as described in Section 3. The quality and precision of this transform, its stationary curve, and its transport are crucial. It turns out that existing intrinsic symmetry methods typically do not yield maps of sufficient quality for our problem (Figure 8).

In this section, we present a simple novel semi-automatic method for construction of generalized bilateral intrinsic symmetries on genus zero surfaces, and a simple embedding method to construct extrinsic symmetry maps on surfaces of arbitrary genus.

### 5.1 Intrinsic symmetry by fixed-point circle

In this section, we assume our surface $M$ is of genus zero.

Our algorithm requires a small number of user-defined correspondences; from these correspondences, it interpolates a smooth symmetry map and provides a stable set of inputs for our field construction algorithm: reliable orbits, smooth stationary line, and robust transport maps.

We have tested several existing automatic algorithms for intrinsic symmetry detection. While on a number of models sufficiently close to extrinsically symmetric or perfect isometry these provide good quality symmetry maps, we found that for many shapes with clear intrinsic symmetry, we could not get orbits and/or stationary lines and and/or transport maps with sufficiently high quality suitable for our algorithm. Even with manual correspondences existing manifold interpolation methods often failed to produce an acceptable map. We compare to existing work in greater detail at the end of this section.

**Representing a generalized reflection as a linear reflection** Our method is motivated by Corollary 3. We compute a generalized reflection $g : \mathbf{M} \to \mathbf{M}$, given as input several pairs of symmetric landmarks (correspondences) $(p_i, q_i) \subset M \times M, i = 1, .., n$. We compute it as a composition of a map $\phi : M \to \widehat{\mathbb{C}}$ to the extended complex plane and a linear reflection in the plane: if we denote $\sigma(x + \mathrm{i}y) = x - \mathrm{i}y$ (the reflection w.r.t. the real axis) then, $\sigma = \phi \circ g \circ \phi^{-1}$, and equivalently,

$$g = \phi^{-1} \circ \sigma \circ \phi. \qquad (6)$$

This point of view has several advantages: 1) it provides a (global) linear representation of the generalized reflection as a simple linear reflection $\sigma$; 2) it provides an analytic representation of the stationary curve $\mathrm{Im}(z) = 0$; and 3) it allows defining the operator $Dg$ (from which we extract the transport $R^g$) in the discrete case in a natural way, as we explain below.

Note that using a per-vertex definition of a map $g : M \to M$, mapping vertices to points on triangles, it is not trivial to define $Dg$ or extract the stationary curve $M(g) = \{g(p) = p\}$ in a robust way numerically.

From the purely topological point of view, there is much freedom in choosing $g, \phi$ that satisfies eq. (6), even when given a set of landmarks $\{p_i, q_i\}$. To restrict the space of possible maps, we add a natural constraint that if there is an *isometric* reflection that interpolates the given landmarks, our map will reproduce it. [Kim et al. 2010] observe that if the surface is mapped conformally to the extended plane $\varphi : M \to \widehat{\mathbb{C}}$, then a perfect intrinsic symmetry $g : M \to M$ has to be an anti-Möbius map over $\widehat{\mathbb{C}}$, i.e., $\varphi \circ g \circ \varphi^{-1} = \widetilde{m}$, where $\widetilde{m}(z) = (a\overline{z} + b)/(c\overline{z} + d)$, for some $a, b, c, d \in \mathbb{C}$ with $ad - bc \neq 0$. To obtain a symmetry map for a set of landmarks mapped to $\widehat{\mathbb{C}}$, [Kim et al. 2010] fits an anti-Möbius transform and uses GH-type (Gromov-Hausdorff) interpolation. However, as the surface deviates significantly from being perfectly intrinsically symmetric this approach suffers from three main drawbacks: 1) it results in non-continuous (and not smooth) symmetry maps; 2) the least-squared fitted anti-Möbius $\widetilde{m}$ do not necessarily satisfy $\widetilde{m}^2 = Id$, nor will it generically have a curve stationary set; and 3) it will not get us the desired representation (6).

We propose a new approach with two key ingredients. First, we use anti-involutions, rather than general anti-Möbius transformations, a

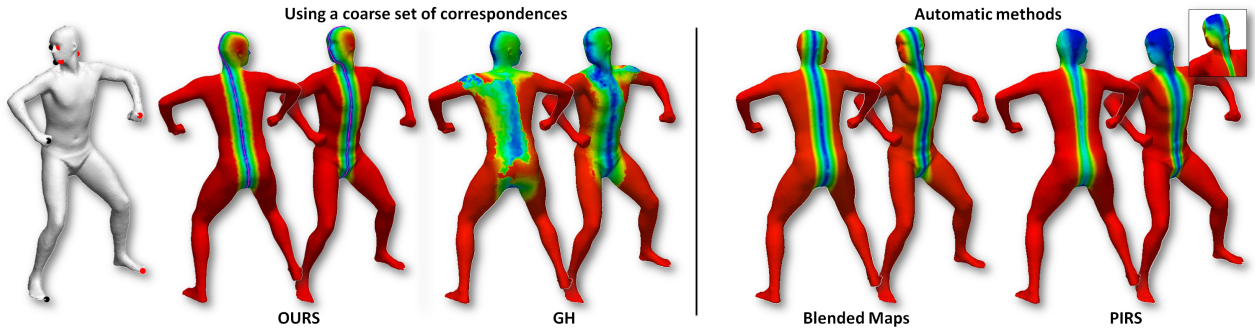


**Figure 8:** *Symmetry detection methods on a human with non-isometric deformation of the neck. On right we show automatic methods: Blended intrinsic maps [Kim et al. 2011], and Partial Intrinsic Reflectional Symmetry (PIRS) [Xu et al. 2009]; note that these erroneously extrapolate the dominant body symmetry to the head. On left, we use a set of coarse correspondences (shown as black and red dots on the leftmost figure); we show GH-type interpolation [Mémoli and Sapiro 2004; Bronstein et al. 2006], side by side with our generalized reflection map. Our map is smooth and have analytic representation of the symmetry stationary curve.*

subset of the anti-Möbius transformations that satisfy $\widehat{m}^2 = Id$ [Schwerdtfeger 1979] which is consistent with our definition of generalized reflection (Definition 1). An important property of anti-involutions is they represent an inversion w.r.t. a circle which is their stationary set $\widehat{m}(z) = z$ (consistent with Proposition 2 and Corollary 3). This stationary circle is our initial guess of the stationary curve. Second, we will define our diffeomorphism $\phi$ using a *smooth* perturbation of the anti-involution. In particular our algorithm consists of the following steps:

1. Calculate the conformal map $\varphi : M \to \widehat{\mathbb{C}}$ of the genus zero mesh to the plane;
2. Transform the landmarks to the extended complex plane $z_i = \varphi(p_i)$, $w_i = \varphi(q_i)$, $i = 1, .., n$;
3. Fit (in the least-squares sense) an anti-involution $\widehat{m}(z_i) \approx w_i$ to the input landmarks;
4. Extract the stationary circle $C$ of the anti-involution defined by $\widehat{m}(z) = z$;
5. Map via a Möbius transformation $m(z)$ the circle $C$ to the real axis;
6. Extract optimal (in the least-squares sense) symmetric configuration of the landmarks $m(z_i), m(w_i)$ w.r.t. the real axis;
7. Use smooth deformation (thin-plate splines) $\psi$ to move the landmarks to their optimal symmetric configurations $\psi(m(z_i)) = \overline{\psi(m(w_i))}$;
8. The final map is $\phi = \psi \circ m \circ \varphi$.

The first and second stages are performed similarly to the algorithm of [Kim et al. 2010].

*Step 3–4.* To find the anti-involution Möbius $\widehat{m}(z)$ minimizing the deviation of $\widetilde{m}(z_i)$ from $w_i$, $i = 1..n$ we use the following lemma:

**Lemma 7.** *An anti-Möbius transformation* $\widetilde{m}(z) = \frac{a\overline{z}+b}{c\overline{z}+d}$ *is an anti-involution iff the matrix of coefficients* $\begin{pmatrix} c & d \\ -a & -b \end{pmatrix}$ *is hermitian, that is* $d = -\overline{a}$, *and* $c, b \in \mathbb{R}$.

For a proof see [Schwerdtfeger 1979] (page 79). Therefore, an anti-involution can be written as $\widehat{m}(z) = (a\overline{z}+b)/(c\overline{z}-\overline{a})$, where $b, c \in \mathbb{R}$. We fit an anti-involution to the symmetry data by multiplying both sides of the equations $\widehat{m}(z_i) = w_i$ by $c\overline{z_i} - \overline{a}$, and solve in the least-squares sense the resulting linear equations:

$$cw_i\overline{z}_i - \overline{a}w_i - a\overline{z}_i - b = 0\ ,\ i = 1, .., n.$$

One delicate point is that the stationary circle of an anti-involution can be imaginary and therefore will not form a "real" circle in $\mathbb{C}$. This would happen if in solving the above equations our solution gives $|a/c|^2 + b/c < 0$. However, this means that the deviation from isometry is extreme and we did not observe this problem in practice.

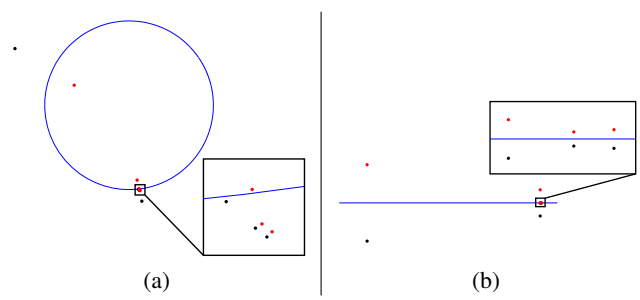


**Figure 9:** *Steps 4-7 in the intrinsic symmetry fit of the model from Figure 8: (a) shows fitting the best circle to a set of symmetric landmarks (red and black). Note how the landmarks in the head area (blow-up figure) are not conforming with the global symmetry. (b) shows the points after a smooth map is applied to bring them to be perfect reflectional symmetric w.r.t. the real axis; the head points are also symmetrized, as the blow-up figure shows.*

*Step 5.* Our next step is to find a Möbius transformation taking the circle stationary set of the anti-involution $\widehat{m}$ to the real axis. The stationary circle of $\widehat{m}$ has the explicit equation $|z - z_0| = r^2$, where $z_0 = a/c$, and $r^2 = |a/c| + b/c$. Figure 9 (a) shows the circle $C$ for the example in Figure 8 (the symmetric landmarks are shown in red and black, consistently with the left image in Figure 8). Note how the landmarks on the head (shown in blow-up figure) do not agree with the global intrinsic symmetry as revealed by the circle.

For the Möbius transform mapping it to the real axis there are three degrees of freedom to set. We simply take three equal distant points $t_1, t_2, t_3$ on the circle $C$ and find the Möbius transform $m$ by solving a system of equations of the type $at_i + b = (ct_i + d)x_i$, $i = 1, 2, 3$, where $x_i \in \mathbb{R}$ are three points on the real axis.

*Step 6–7.* Next, we map the landmarks $z_i, w_i$ with $m$ and extract $L^2$ symmetric optimal points $s_i \in \mathbb{C}$, $i = 1, .., n$ such that $s_i = \operatorname{argmin}_z \left[|s_i - m(z_i)|^2 + |\overline{s_i} - m(w_i)|^2\right]$. It is not hard to see that the minimizer is $s_i = (m(z_i) + \overline{m(w_i)})/2$. Finally, we find a thin-plate spline $\psi : \mathbb{R}^2 \to \mathbb{R}^2$ such that $\psi(m(z_i)) = s_i$, and

$\psi(m(w_i)) = \overline{s}_i$. Our final map is $\phi = \psi \circ m \circ \varphi$, the symmetry map in these new coordinates is $z \mapsto \overline{z}$ and the stationary curve is $\mathrm{Im}(z) = 0$. Figure 9(b) shows the landmarks $z_i, w_i$, after applying $\psi \circ m$ to it; note that they are now perfectly symmetric w.r.t. the real axis. Figure 8 shows (in purple) the stationary curve $\mathrm{Im}(z) = 0$ back-projected to the original model.

**Computational complexity.** The algorithm for finding $\phi$, and $g = \phi^{-1} \circ \sigma \circ \phi$ requires solving one sparse linear system in the size of the mesh for the discrete harmonic part (standard cotangent Laplacian) of the uniformization, followed by linear time computation of its conjugate. The next steps of the algorithm are negligible computationally and require solving two linear systems in the order of number of landmarks (we typically used 5 to 10 landmarks).

**Evaluation of orbits, stationary lines and transport.** Function $\phi$ provides a parametrization for $M$, which can be evaluated in both directions: given a point $p \in M$, its symmetric mate $g(p)$ is evaluated using Equation 6. The stationary line $M(g)$ is simply the pre-image $\phi^{-1}(\ell)$of the real line $\ell = \{\mathrm{Im}(z) = 0\}$. More specifically, given a triangle $t_i$ of $M$, let $\bar{t}_i$ be $\phi(t_i)$. Triangle $t_i$ intersects the stationary line if and only if $\bar{t}_i$ intersects the real axis; and the linear segment of stationary line corresponding to $t_i$ has endpoints at $\phi^{-1}(a)$ and $\phi^{-1}(b)$, where $a$ and $b$ are the intersections of edges of $\bar{t}_i$ with the real axis.

**Evaluation of transport $R^g$.** In principle, the differential $Dg$ and, as a consequence, $R^g$ can be obtained by computing the gradient of $g$ numerically; while the maps we construct in the intrinsic case are quite smooth, we found that the following method provides the most robust results. Let $\zeta, \xi : M \longrightarrow \mathbb{R}$ be two scalar functions such that $\zeta$ is symmetric and $\xi$ is antisymmetric, i.e., $\zeta(p) = \zeta(g(p))$ and $\xi(p) = -\xi(g(p))$. By chain rule,

$$\begin{aligned} \nabla\zeta(p) &= \nabla\zeta(g(p))Dg_p \qquad \text{and} \\ \nabla\xi(p) &= -\nabla\xi(g(p))Dg_p. \end{aligned} \tag{7}$$

where $Dg_p$ is the differential of $g$ computed at $p$. System (7) allows us to solve for $Dg$ at any point, if the surface gradients of $\zeta$ and $\xi$ are computed everywhere. $R^g$ is then obtained from $Dg$ as described in Section 3. In our case, we choose $\zeta(p) = \mathrm{Re}(\phi(p))$ and $\xi(p) = \mathrm{Im}(\phi(p))$.

**Comparison to previously proposed methods for intrinsic symmetry detection.** A simple example is shown in Figure 8. This model of a human (SCAPE dataset [Anguelov et al. 2005]) has a dominant extrinsic reflection symmetry (neck down) while the head is rotated to the right. The deformation at the neck area involves a considerable metric distortion with respect to the rest pose, which makes recognizing head symmetry difficult for a global method. This problem is reinforced since the head has pretty good continuous rotational symmetry excluding negligible (in terms of area) features like ears and nose. Figure 8 shows the result of two automatic state-of-the-art-methods: PIRS [Xu et al. 2009], and Blended Intrinsic Maps [Kim et al. 2011]. PIRS is a particularly suitable algorithm for extracting stationary lines. However, as it is based on aggregating votes, the dominant extrinsic symmetry of the surface in this case overrides the symmetry of the head. Blended Intrinsic Maps also use a global deformation energy and therefore "extrapolate" the dominant extrinsic symmetry to the head.

We also compare to a method in the spirit of Gromov-Hausdorff (GH) [Bronstein et al. 2006; Mémoli and Sapiro 2004], where geodesic distances to the known landmark points are used to define feature vectors and closest point in the corresponding feature space defines the correspondences. Using five pairs of symmetric points (shown at the left of Figure 8) with GH-type interpolation produces a reasonable result in certain areas (like the head and the lower torso) where there are relatively many landmarks and/or geodesic distances reliably detect the symmetry transform. However, in vicinity of intrinsically distorted areas, where geodesics are not reliable, this method tends to produce results of lower quality compared to ours (e.g., the neck and left shoulder area). Another drawback of this type of methods in our context is that the correspondence mapping is not necessarily smooth or even continuous, and it hard to reliably extract the field transport from it. Lastly, given the set of correspondences it is also not trivial to extract the stationary line to be a closed curve, see for example the bottom and neck area again where any such curve is likely to be cut.

### 5.2 Extrinsic symmetry through embedding

In the extrinsic case, a reflection mapping $g$ is a simple orthogonal reflection about a plane $\Pi$ in 3D space, its stationary set $M(g)$ coincides with $M \cap \Pi$, and its differential $Dg = R^g$ is $g$ itself.

For this simpler case, we have combined two existing algorithms to find the reflection $g$ and the stationary set $M \cap \Pi$ in a robust way. Following [Osada et al. 2002] we have built extrinsic symmetry invariants at every point and used it to define a symmetry correspondence matrix, similar to [Lipman et al. 2010]. The stationary plane was found by averaging the points' 3D coordinates over fuzzy orbits, and the final reflection $g$ was set according to that plane. Let us briefly elaborate on each step.

**Extrinsic symmetry invariant histograms.** For each triangle $t$ of $M$, we compute the Euclidean distances from its centroid $c_t$ to the centroids of all other triangles of $M$, and we build a histogram of these distances using $b$ bins. (We have used $b = 50$ for all our results.) We denote by $\Psi(t) \in \mathbb{R}^b$ the histogram vector for triangle $t$. Next, we fill a symmetry correspondence matrix $S$, by setting $S_{i,j} = m - d(\Psi(t_i), \Psi(t_j))$, where $d$ is the Euclidean 2-norm, and $m = \max_{i,j} d(\Psi(t_i), \Psi(t_j))$. We make sure $S$ is sparse by keeping only 50 biggest values at each row and zero everywhere else. Finally, we normalize each row of $S$ to make it row-stochastic. The $i$-th row of $S$ provides the fuzzy orbit of triangle $t_i$.

**The symmetry plane and stationary curve $M(g)$.** Let $C_t \in \mathbb{R}^{n\times 3}$, where $n$ is the number of triangles, be the matrix of all centroids' 3D coordinates. The product $SC_t$ averages the 3D coordinates over the the orbit and therefore projects the points onto the stationary set (in 3D) (see [Lipman et al. 2010]). We fit a plane $\Pi$ using Principal Component Analysis (PCA). The stationary line is extracted simply by collecting the triangles of $M$ that intersect $\Pi$.

**Field transport $R^g$.** We perform a change of coordinates that places the origin in $\Pi$ and aligns the $z$ axis with the least significant direction yield by the PCA. Then we estimate $Dg = R^g$ by means of the procedure that we use for intrinsic symmetries, by setting $\zeta(p) = p_x$ and $\xi(p) = p_z$, where $p = (p_x, p_y, p_z)$ is a generic point of $M$.

## 6 Fields that are symmetric with respect to sets of transforms

Next we explain how the theory and algorithms of the previous sections apply in the case when a shape has multiple symmetries. It is natural to define a field to be symmetric with respect to a set $G$ of transforms, if it is symmetric with respect to all elements of $G$. For example, $G$ may consists of reflections about two or more planes of symmetries. The input to our algorithm requires orbits, stationary lines, and transport maps.

The orbit for a point $p$ with respect to a set $G$ are simply the union of the orbits for individual transforms in $G$ (we want to symmetrize

our field at a point $p$ with field values at all points $g(p)$, $g \in G$).

Similarly, the stationary set of interest is the union of the stationary set of individual elements of $G$. Note that if a point $p$ is in two stationary sets $M(g_1)$ and $M(g_2)$, there are two distinct alignment requirements for the field. If the stationary directions of $R^{g_1}(p)$ an $R^{g_2}(p)$ are orthogonal, then both have to be satisfied at once, which may not always be possible, in which case the best we can do is to minimize the deviation from symmetry.

Defining the transport map requires a consistency condition between symmetry maps: there may be two maps $f$ and $h$ mapping a point $p_1$ to the same point $p_2 = f(p_1) = h(p_2)$. We want the transports defined by $f$ and $g$ to be the same, so we require

$$R^f v = R^h v, \text{ at } p_1, \text{ for any } f \text{ and } h \text{ with } f(p_1) = h(p_1). \quad (8)$$

Observe that in this case, $h^{-1} \circ f$ is a non-identity transform with $p_1$ as a stationary point. So the field at $p_1$ and the map $R^{h^{-1}} R^f$ has to satisfy Proposition 6. In the cases of primary interest to us (sets of generalized reflections) the compositions of distinct reflections have, in general, only isolated stationary points which have little effect on the overall field behavior.

For all other points $p$, for each $q$ in the orbit of $\mathcal{O}(p)$, there is a unique $g$ mapping $p$ to $q$, which defines the transport of the field from $p$ to $q$ unambiguously.

**Rotations.** A rigid rotation can be computed as a composition of two reflections, and in general it has only isolated stationary points on a surface. It is natural to generalize the rotations as compositions of two reflections with isolated stationary points. This allows direct extension of all constructions we need to the case of sets including such generalized rotations.

## 7 Results

We have tested our methods on a range of geometric objects with approximate intrinsic and extrinsic symmetries (see Table 1). In this section, we describe our findings, discuss alternative approaches and design choices we made. Our algorithms are used to detect symmetries and generate symmetric $N$-symmetry fields. We use existing algorithms for field rendering ([Nieser et al. 2012]) and field-aligned parametrization and quadrangulation (MIQ, [Bommes et al. 2009]). Note that we do *not* modify MIQ process for fitting the parametrization to a field: this would be desirable, e.g., to force the stationary line to lie exactly on an integer parametric line. As our focus is on computing symmetric fields, we leave full integration of symmetry into all stages of the quadrangulation pipeline as future work.

**Extrinsic and intrinsic symmetry detection performance.** The method for intrinsic symmetry detection required between 5 and 10 landmarks (as specified in Table 1), and typical running times were between one and five minutes. Comparison with other methods have been discussed in Section 5 and in particular Figure 8. The method for extrinsic symmetry detection is about five times slower (as it requires creating histograms of all pair distances), but it is fully automatic and it works for objects of any genus. On objects where both methods can be applied, results are overall similar (see Figure 10).

**Stationary line constraints.** We found that stationary line constraints play an essential role in the quality of the results; Figure 12 compares the result of our algorithm with the result we obtain with no stationary line constraints. Note that without this constraint, although generally symmetric the field is not "parallel" along the stationary curve.

**Figure 10:** *Comparison of extrinsic (left) and intrinsic (right) symmetry detection on an extrinsically symmetric model. There are small differences in cone placements, but the field alignment is similar.*

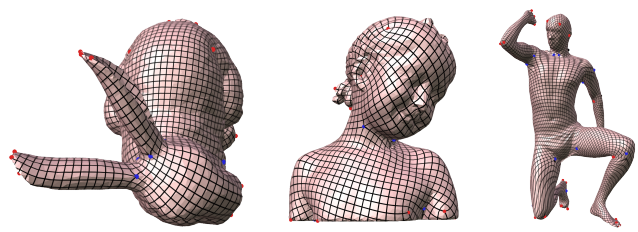

**Figure 11:** *Imposing symmetric cones is not sufficient, generally, to produce symmetric fields.*

**Effects of smoothness parameter.** Our algorithm has three parameters: two parameters defining the features used in the mixed-integer field optimization, and the smoothness parameter $\alpha$. The feature-detection parameters are the same as in [Bommes et al. 2009], and their effect is limited to changing the number of constrained faces. Figure 7 shows dependence of the results on the choice of $\alpha$: the higher $\alpha$ is, the sharper the transition between features and symmetrized areas; as a result the field's smoothness decreases and additional cones may appear.

**Multiple symmetries.** Figure 13 shows experiments with multiple symmetry objects. For multiple stationary lines, the field is fixed on their union. Note how cones tend to respect symmetry relations in this case as well.

**Comparisons.** As there are no direct analogs of our method, we compare to a "naive" method described below and a non-symmetry-aware method of [Bommes et al. 2009].

*"Naive" method.* Once the symmetry map is available, a naive approach would be to first compute a non-symmetric field (with MIQ); then reflecting the cone singularities found on one side of the stationary line to their symmetric mates; and finally computing a field constrained to such singularities. As shown in Figure 11, while this may work in some cases (bunny), in general the field fails to follow the stationary line (e.g. Bimba model) and it may also introduce very large distortions (e.g. the right leg of the human model).

*Mixed-integer field optimization.* In most cases, "symmetry-unaware" mixed integer algorithm would not produce symmetric results, as its global behavior depends strongly on the local feature symmetry. Only if the feature constraints of the MIQ are well dis-

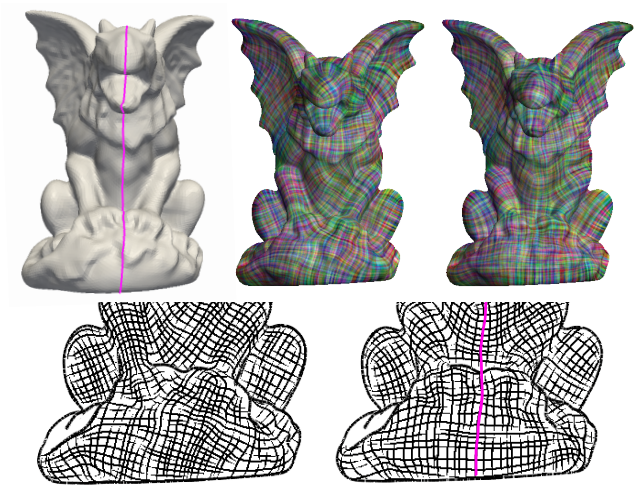

**Figure 12:** *A field symmetrized without (center) and with (right) stationary line constraints. The field in the middle/left image is overall symmetric but it does not follow the stationary line.*

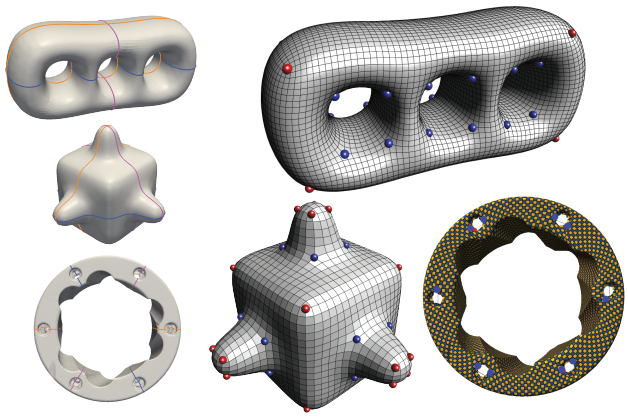

**Figure 13:** *Stationary lines and quadrangulations for a few objects with several bilateral symmetries.*

tributed and symmetric, a symmetric field is produced.

Figures 1 and 14 show our results compared with [Bommes et al. 2009]. In most cases, we observe a substantial improvement in visual quality: in the standing human model, note how our method produces fields following the line corresponding to sagittal plane even under a strong non-isometric deformation of the neck and head. Our method also preserves a horizontal field on the chest. Similar behavior can be seen in the Busto and Igea models, where MIQ-placed singularities asymmetrically at small details of the face, thus pushing the field away from respecting symmetry. On objects with an extrinsic global symmetry, like the lion and fertility models, plain MIQ yields satisfactory results, yet our method still improves the symmetry awareness of the resulting field. In the Igea model, our algorithm preserves global symmetry while adapting to local non-symmetric features like the asymmetric broken piece from one cheek. Note how our method manages symmetry also in complicated models like the octopus (small picture shows the symmetry line), and in open meshes like the lion (open at the neck). In the teddy bear we show an example of 6-symmetry field (MIQ version is obtained with a straightforward variant of the MIQ field optimization).

| Model: | # faces | $\alpha$ | int/ext | # lmks | # cones |
|---|---|---|---|---|---|
| bimba | 100,000 | 0.15 | I | 9 | 97 / 87 |
| bumpy-cube | 34,754 | 0.25 | E | - | 56 / 56 |
| busto | 50,930 | 0.30 | I | 8 | 28 / 46 |
| egea | 30,000 | 0.35 | E | - | 26 / 32 |
| fertility | 46,388 | 0.10 | E | - | 38 / 44 |
| gargoyle | 54,000 | 0.25 | E | - | 94 / 92 |
| holes3 | 28,800 | 0.15 | E | - | 32 / 16 |
| knelt human | 40,000 | 0.10 | I | 5 | 56 / 64 |
| lion | 66,696 | 0.25 | E | - | 66 / 66 |
| Max Planck | 54,000 | 0.20 | E | . | 20 / 33 |
| octopus | 10,000 | 0.25 | I | 7 | 76 / 74 |
| ogre | 52,306 | 0.30 | I | 10 | 238 / 265 |
| rolling | 20,000 | 0.12 | E | - | 27 / 33 |
| stand. human | 24,978 | 0.10 | I | 8 | 49 / 57 |
| teddy | 25,290 | 0.10 | I | 4 | 42 / 34 |

**Table 1:** *Statistics of datasets and results. We report: number of faces in the input; value of parameter $\alpha$ used; whether it has intrinsic or extrinsic symmetry; number of landmarks used for detecting intrinsic symmetry; number of cone singularities in the output (our algorithm / MIQ).*

## 8 Concluding remarks

We introduced a new approach to symmetry-aware field design on surfaces. Our key idea is to incorporate symmetry averaging of field values over symmetry orbits with existing mixed-integer field generation techniques. This required several steps: 1) introducing general non-isometric reflections, and developing the formulations for field transport and symmetric averaging; 2) developing methods to compute symmetric maps, transport and stationary sets on shapes deviating from perfect isometry; and 3) incorporating the symmetric averaging operator into the Mixed-Integer framework. Based on experiments on a variety of models, we believe the introduced algorithm can significantly improve visual aesthetics and symmetry-awareness of $N$-symmetry fields on models.

A significant limitation of our intrinsic symmetry construction method is the restriction to genus-zero objects: extension of this technique to higher-genus objects is nontrivial and is an important direction for future work. $N$-symmetry fields are often a building block in other types of geometry processing algorithms; symmetry also needs to be integrated with other stages of these algorithms (e.g., quadrangulation or texture generation).

## Acknowledgments

We wish to thank Ashish Myles for kindly providing his implementation of the MIQ framework.

## References

ANGUELOV, D., SRINIVASAN, P., KOLLER, D., THRUN, S., RODGERS, J., AND DAVIS, J. 2005. Scape: shape completion and animation of people. *ACM Trans. Graph. 24* (July), 408–416.

BOMMES, D., ZIMMER, H., AND KOBBELT, L. 2009. Mixed-integer quadrangulation. *ACM Trans. Graph. 28*, 3, 77.

BOMMES, D., LEMPFER, T., AND KOBBELT, L. 2011. Global structure optimization of quadrilateral meshes. *Comput. Graph. Forum 30*, 2, 375–384.

BRONSTEIN, A. M., BRONSTEIN, M. M., AND KIMMEL, R. 2006. Generalized multidimensional scaling: a framework for

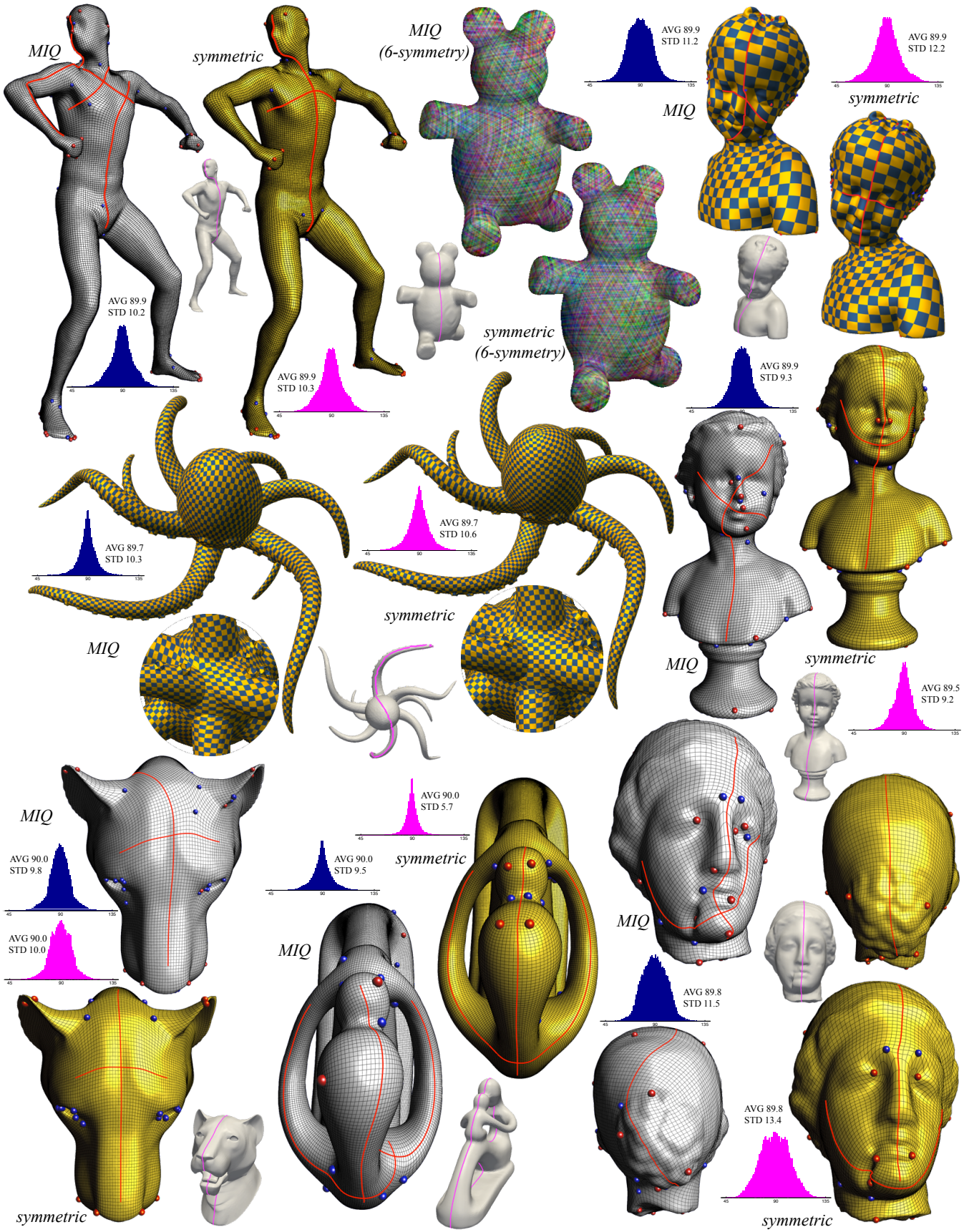


**Figure 14:** *Comparisons of our results with respect to plain MIQ. Histograms show angle distribution in elements of quadrangulation (in degrees - AVG: average - STD: standard deviation). Line of symmetry depicted in magenta on the small figures. Rendering with: MIQ remeshing (standing human, busto, lion, fertility, egea); checkerboard texturing (bimba, octopus); LIC from [Palacios and Zhang 2011] (teddy bear). Red/blue bullets represent singularities field singularities with positive/negative index.*

isometry-invariant partial surface matching. *Proc. Natl. Acad. Sci. USA 103*, 5, 1168–1172.

CAILLIERE, D., DENIS, F., PELE, D., AND BASKURT, A. 2008. 3d mirror symmetry detection using hough transform. In *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, IEEE, 1772–1775.

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

FARB, B., AND MARGALIT, D. 2011. *A primer on mapping class groups*. Princeton Univ Press.

GHOSH, D., AMENTA, N., AND KAZHDAN, M. 2010. Closed-form blending of local symmetries. *Computer Graphics Forum 29*, 5, 1681–1688.

GOLOVINSKIY, A., PODOLAK, J., AND FUNKHOUSER, T. 2009. Symmetry-aware mesh processing. *Mathematics of Surfaces XIII*, 170–188.

HERTZMANN, A., AND ZORIN, D. 2000. Illustrating smooth surfaces. In *Proceedings of SIGGRAPH 2000*, 517–526.

KÄLBERER, F., NIESER, M., AND POLTHIER, K. 2007. QuadCover: Surface Parameterization using Branched Coverings. *Computer Graphics Forum 26*, 3, 375–384.

KAZHDAN, M., AMENTA, N., GU, S., WILEY, D., AND HAMANN, B. 2009. Symmetry restoration by stretching. In *Canadian Conference on Computational Geometry*, Citeseer.

KIM, V. G., LIPMAN, Y., CHEN, X., AND FUNKHOUSER, T. 2010. Mbius Transformations For Global Intrinsic Symmetry Analysis. *Computer Graphics Forum 29*, 5, 1689–1700.

KIM, V. G., LIPMAN, Y., AND FUNKHOUSER, T. 2011. Blended intrinsic maps. *ACM Trans. Graph. 30* (Aug.), 79:1–79:12.

KOSZUL, J. 1965. *Lectures on groups of transformations*, vol. 32 of *Lectures on Mathematics*. Tata Institute of Fundamental Research, Bombay, India.

KRAEVOY, V., AND SHEFFER, A. 2004. Cross-parameterization and compatible remeshing of 3d models. *ACM Transactions on Graphics (Proc. SIGGRAPH 2004)*.

LAI, Y., JIN, M., XIE, X., HE, Y., PALACIOS, J., ZHANG, E., HU, S., AND GU, X. 2010. Metric-driven rosy field design and remeshing. *Visualization and Computer Graphics, IEEE Transactions on 16*, 1, 95–108.

LIPMAN, Y., CHEN, X., DAUBECHIES, I., AND FUNKHOUSER, T. 2010. Symmetry factored embedding and distance. In *ACM SIGGRAPH 2010 papers*, ACM, 1–12.

MÉMOLI, F., AND SAPIRO, G. 2004. Comparing Point Clouds . In *Proceedings Symposium on Geometry Processing 2004*, Eurographics, 33–42.

MITRA, N., GUIBAS, L., AND PAULY, M. 2006. Partial and approximate symmetry detection for 3d geometry. *ACM Transactions on Graphics (TOG) 25*, 3, 560–568.

MITRA, N., GUIBAS, L., AND PAULY, M. 2007. Symmetrization. *ACM Transactions on Graphics 26*, 3.

MONTGOMERY, D., AND ZIPPIN, L. 1955. *Topological transformation groups*, vol. 1. Interscience Publishers New York.

NIESER, M., PALACIOS, J., POLTHIER, K., AND ZHANG, E. 2012. Hexagonal global parameterization of arbitrary surfaces. *Visualization and Computer Graphics, IEEE Transactions on 18*, 6 (june), 865 –878.

OSADA, R., FUNKHOUSER, T., CHAZELLE, B., AND DOBKIN, D. 2002. Shape distributions. *ACM Trans. Graph. 21* (October), 807–832.

OVSJANIKOV, M., SUN, J., AND GUIBAS, L. 2008. Global intrinsic symmetries of shapes. *Computer graphics forum 27*, 5, 1341–1348.

OVSJANIKOV, M., MÉRIGOT, Q., MÉMOLI, F., AND GUIBAS, L. 2010. One Point Isometric Matching with the Heat Kernel. *Computer Graphics Forum 29*, 5, 1555–1564.

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

PALACIOS, J., AND ZHANG, E. 2011. Interactive visualization of rotational symmetry fields on surfaces. *IEEE Transactions on Visualization and Computer Graphics 17*, 7 (July), 947–955.

PENG, C.-H., ZHANG, E., KOBAYASHI, Y., AND WONKA, P. 2011. Connectivity editing for quadrilateral meshes. *ACM Trans. Graph. 30* (Dec.), 141:1–141:12.

PODOLAK, J., SHILANE, P., GOLOVINSKIY, A., RUSINKIEWICZ, S., AND FUNKHOUSER, T. 2006. A planar-reflective symmetry transform for 3d shapes. *ACM Transactions on Graphics 25*, 3, 549–559.

PODOLAK, J., GOLOVINSKIY, A., AND RUSINKIEWICZ, S. 2007. Symmetry-enhanced remeshing of surfaces. In *Proceedings of the fifth Eurographics symposium on Geometry processing*, Eurographics Association, 235–242.

PRAUN, E., SWELDENS, W., AND SCHRÖDER, P. 2001. Consistent mesh parameterizations. *Proc. of SIGGRAPH 2001*.

RAVIV, D., BRONSTEIN, A. M., BRONSTEIN, M. M., AND KIMMEL, R. 2007. Symmetries of non-rigid shapes. In *Proc. Non-rigid Registration and Tracking (NRTL) workshop. See Proc. of International Conference on Computer Vision (ICCV)*.

RAVIV, D., BRONSTEIN, A., BRONSTEIN, M., AND KIMMEL, R. 2010. Full and partial symmetries of non-rigid shapes. *International journal of computer vision 89*, 1, 18–39.

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

RAY, N., VALLET, B., LI, W., AND LÉVY, B. 2008. N-Symmetry Direction Field Design. *ACM Trans. Graph. 27*, 2.

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

SCHREINER, J., ASIRVATHAM, A., PRAUN, E., AND HOPPE, H. 2004. Inter-surface mapping. *ACM Transactions on Graphics (Proc. SIGGRAPH)*.

SCHWERDTFEGER, H. 1979. *Geometry of complex numbers: circle geometry, Moebius transformation, non-euclidean geometry*. Dover Books on Mathematics Series. Dover.

TARINI, M., PUPPO, E., PANOZZO, D., PIETRONI, N., AND CIGNONI, P. 2011. Simple quad domains for field aligned mesh parametrization. *ACM Trans. Graph. 30* (Dec.), 142:1–142:12.

XU, K., ZHANG, H., TAGLIASACCHI, A., LIU, L., LI, G., MENG, M., AND XIONG, Y. 2009. Partial intrinsic reflectional symmetry of 3d shapes. *ACM Transactions on Graphics (TOG) 28*, 5, 138.