

Partial Intrinsic Reflectional Symmetry of 3D Shapes

Kai Xu^{*‡*} Hao Zhang^{**} Andrea Tagliasacchi^{*} Ligang Liu^{†*} Guo Li[†] Min Meng[†] Yueshan Xiong[‡]
^{*}Simon Fraser University [†]Zhejiang University [‡]National University of Defense Technology

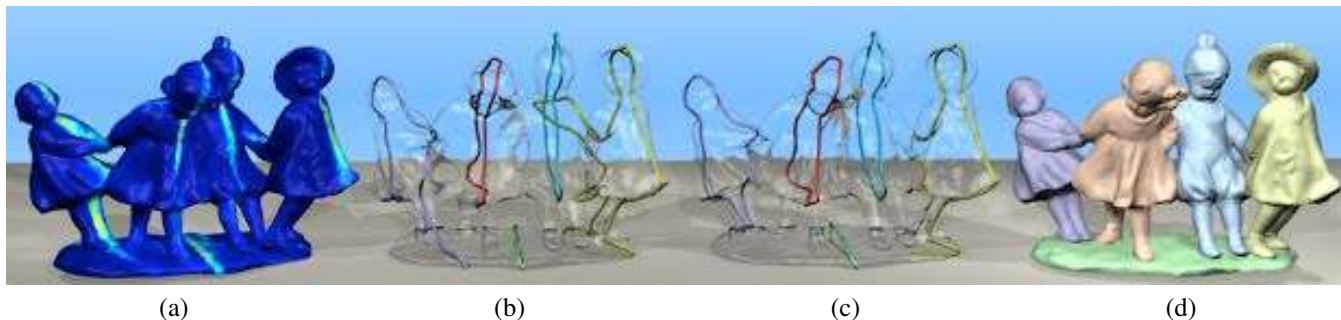


Figure 1: Given a closed 2-manifold mesh, we compute a scalar field (a), which accentuates the axes of prominent, partial intrinsic reflectional symmetries. The top few (closed) Voronoi boundaries (b) between symmetric point pairs, as induced by the scalar field, can be imperfect. We develop an iterative refinement scheme to extract the final set of *intrinsic reflectional symmetry axes* or IRSAs (c), which can be open curves. Incorporating symmetry cues offered by IRSAs into a conventional mesh segmentation scheme leads to highly semantic results (d).

Abstract

While many 3D objects exhibit various forms of global symmetries, prominent intrinsic symmetries which exist only on parts of an object are also well recognized. Such partial symmetries are often seen as more natural than a global one, even when the symmetric parts are under complex pose. We introduce an algorithm to extract *partial intrinsic reflectional symmetries* (PIRS) of a 3D shape. Given a closed 2-manifold mesh, we develop a voting scheme to obtain an intrinsic reflectional symmetry axis (IRSA) transform, which is a scalar field over the mesh that accentuates prominent IRSAs of the shape. We then extract a set of explicit IRSA curves on the shape based on a refined measure of local reflectional symmetry support along a curve. The iterative refinement procedure combines IRSA-induced region growing and region-constrained symmetry support refinement to improve accuracy and address potential issues arising from rotational symmetries in the shape. We show how the extracted IRSA curves can be incorporated into a conventional mesh segmentation scheme so that the implied symmetry cues can be utilized to obtain more meaningful results. We also demonstrate the use of IRSA curves for symmetry-driven part repair.

1 Introduction

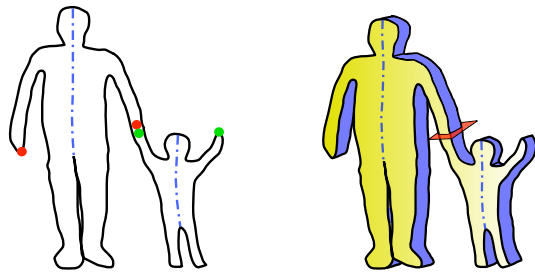
The ubiquity of symmetry in nature and man-made artifacts has been well-documented and the study of symmetry, along with its

implications, has appeared in diverse disciplines including mathematics, arts, biology, astronomy, mechanical design, architecture, and psychology [Weyl 1983; Stewart and Golubitsky 1992; Leyton 2001]. Symmetry has been shown to play a crucial role in human cognition [Leyton 1992] and in particular, the Gestalt law of symmetry attributes symmetry as one of the fundamental principles of perception [Köhler 1929]. It then comes as no surprise that several algorithms in shape processing have exploited shape symmetry recently. These include compression [Simari et al. 2006] and remeshing [Podolak et al. 2007] which utilize symmetry-induced redundancy, segmentation via grouping by planar-reflective symmetry transform [Podolak et al. 2006], shape editing [Martinet et al. 2006], object retrieval [Kazhdan et al. 2004b], as well as model repair [Simari et al. 2006] or beautification [Mitra et al. 2007].

Reflectional symmetry is the most fundamental form of symmetry observed. Robust algorithms for detecting reflection lines or planes which support approximate and partial *extrinsic* symmetries exist [Podolak et al. 2006; Simari et al. 2006]. Symmetries among shape parts can also be discovered via transform-space voting [Mitra et al. 2006], assuming that the symmetric parts only differ by a *rigid* transformation. A general *global intrinsic* symmetry of a shape is defined to be an isometric (geodesic distance preserving) homeomorphism between the shape surface and itself which leaves the shape globally unchanged. So far, methods developed to detect such symmetries [Martinet et al. 2006; Raviv et al. 2007; Ovsjanikov et al. 2008] do not apply to an intrinsically asymmetric shape composed of intrinsically symmetric parts since the symmetry-revealing self-mapping may be necessarily discontinuous, as shown in Figure 2(a). Even when a global intrinsic symmetry does exist, it may not be a natural one — see Figure 2(b).

The examples from Figure 2 reveal that a complex shape may be composed of symmetric sub-parts which themselves possess complex structure, such as a human shape, yet these sub-shapes are often easily identified by a human. Indeed, symmetry seems to play a fundamental role in cognitive grouping and object identification. Instead of grouping the two holding arms of the mother and child together, possibly due to proximity cues or the minima rule [Hoffman and Richards 1984], we find it more natural to separate the two arms and decompose the shape into two symmetric sub-parts.

^{*}Corresponding authors: kaixu@nudt.edu.cn (Kai Xu), haoz@cs.sfu.ca (Hao Zhang), ligangliu@zju.edu.cn (Ligang Liu)



(a) IRSAs (blue) and matching point pairs (red and green). (b) IRSAs and the implied natural segmentation.

Figure 2: Partial vs. global intrinsic symmetries. (a) An asymmetric 2D shape composed of two symmetric parts. The symmetry-revealing self-mapping is necessarily discontinuous, as can be observed from the two matching colored dots. (b) A 3D shape obtained from (a) via extrusion has a global symmetry: front and back. However, the individual symmetries of the mother and child appear to be more natural and trigger a stronger visual response.

Going one step further, it has been argued that it may be possible to recover the *generation history* of a complex shape by modeling the history as a series of symmetry breakings [Leyton 2001].

In this paper, we consider the problem of detecting intrinsic, reflectional symmetries which exist on parts of a 3D shape; we call this *partial intrinsic reflectional symmetry* or PIRS. In particular, if the shape is composed of complex, intrinsically symmetric parts, we seek to recover such composition so as to reveal the underlying semantics of the composite shape. The notion of parts in this context is induced by intrinsic shape symmetry, which is drastically different from low-level geometric notions such as convexity or part salience [Hoffman and Singh 1997]. Without venturing into human cognition, we take a purely geometric approach to tackle our problem and make the following contributions:

- A novel algorithm for computing a scalar field over a closed manifold mesh which accentuates the prominent *intrinsic reflectional symmetry axes*, or IRSAs, of the represented shape. We call the scalar field the *IRSA transform* of the mesh. The voting scheme of Podolak et al. [2006] for planar reflective symmetry transform (PRST) has provided the inspiration. It is adapted to the intrinsic surface setting, with several key challenges specific to this setting addressed by introducing new sampling biases into the voting paradigm.
- An iterative algorithm to extract explicit IRSA curves from an IRSA transform. The IRSA curves serve as the true axes of PIRSeS of the shape for subsequent processing.
- Introduction of PIRS-induced shape semantics into an existing mesh segmentation framework. Compared to the use of local measures by low-level segmentation schemes [Shamir 2006; Chen et al. 2009], intrinsic symmetry operates at a more global and semantic level for shape analysis.
- An algorithm for semantic, symmetry-driven part repair.

Figure 1 shows the IRSA transform (visualized by a color plot of the scalar field), IRSA curves, and a symmetry-aware mesh segmentation (compare with results given in Figure 3) computed on the children model. With the ability to detect PIRSeS on 3D shapes, our algorithm allows any previous symmetry-aware processing scheme, e.g., [Podolak et al. 2007], to be applied to shape parts.

The use of a voting paradigm for the detection of symmetries or self-similarities in shapes or images has seen much success in the



Figure 3: Results of a conventional mesh segmentation scheme [Liu and Zhang 2007] based on the minima rule (use of concavity) are not so meaningful. Left: 5 parts (compare with Figure 1(d) for an IRSA-induced segmentation also having 5 parts). Right: 16 parts.

extrinsic setting [Loy and Eklundh 2006; Mitra et al. 2006; Podolak et al. 2006]. In particular, our voting scheme is inspired by the work of Podolak et al. [2006]. However the most important difference between the two schemes is that our symmetry detection, being intrinsic, tolerates large pose changes, while the one in the work of Podolak et al. [2006] does not. Complexity-wise, any intrinsic approach is naturally expected to be more expensive due to the much higher degree of freedom the axes of PIRSeS of a 3D shape can assume. This leads to a much larger search space, hence more sophisticated search strategies and a more elaborate voting scheme are needed to obtain a feasible solution.

To handle pose, it is possible to apply eigenanalysis [Ovsjanikov et al. 2008] or resort to approximately isometric embeddings such as multi-dimensional scaling [Elad and Kimmel 2001] so that the intrinsic symmetry detection problem becomes an extrinsic one in the embedding space. However, metric distortions in the embeddings are inevitable and they can exaggerate slight asymmetries in the original shape. In particular, these approaches do not perform well for high-genus models such as the children in Figure 1.

2 Related work

Symmetry analysis has been extensively studied in various fields including image processing, visual perception, and computational geometry [Atallah 1985; Wolter et al. 1985; Zabrodsky and Weinshall 1997; Golovinskiy et al. 2007]. It has also been widely considered in geometry processing including shape matching [Bronstein et al. 2006; Gal and Cohen-Or 2006], retrieval [Rustamov 2008], alignment [Bronstein et al. 2007; Chaouch and Verroust-Blondet 2008], remeshing [Podolak et al. 2007], and segmentation [Podolak et al. 2006; Riklin-Raviv et al. 2006; Simari et al. 2006].

Most existing works have focused on detecting and classifying extrinsic symmetries. Zabrodsky and Weinshall [1997] detected approximate symmetry by defining a symmetry measure for a given transformation as the distance of a shape to the closest symmetric shape. Kazhdan et al. [2004b] developed a transform that measures the reflective symmetries of a 2D shape with respect to all possible lines in the plane. This method was extended to work on 3D shapes [Podolak et al. 2006] and was augmented to include spatial distribution of the object's asymmetry [Riklin-Raviv et al. 2007]. Martinet et al. [2006] proposed a method for recovering global symmetry of 3D shapes based on generalized moments, analyzing the extrema and spherical harmonic coefficients. The series of papers of [Kazhdan et al. 2002; Kazhdan et al. 2004a; Kazhdan et al. 2004b] used global symmetry as a shape descriptor for shape analysis and matching. Recently, different schemes have been proposed to detect repeated patterns in 3D geometric data [Pauly et al. 2008; Bokeloh et al. 2009] and 2D line art [Yeh and Mech 2009].

Detecting partial or local extrinsic symmetries is also of interest. Thrun and Wegbreit [2005] introduced a technique for finding local symmetries of objects based on a hierarchical generate-and-test procedure. Simari et al. [2006] proposed an approach capa-

ble of detecting local approximate planar symmetries, defining a shape as a hierarchical union of planar symmetric parts. Mitra et al. [2006] proposed to detect approximate partial symmetries for shapes based on transform-space voting and clustering, while a similar approach [Loy and Eklundh 2006] utilizing Hough transforms had been applied to symmetry detection in images. With partial symmetries in a shape detected, Mitra et al. [2007] sought to deform the shape into an extrinsically symmetric one.

Intrinsic symmetries are defined as invariant under isometric transformations of the shape and are more difficult to represent and compute. All the works to date have been on detecting global intrinsic symmetries. Raviv et al. [2007] proposed a bottom-up method for searching an isometric mapping between a set of sample points and itself over a manifold surface. Observing that the Global Point Signatures (GPS) of [Rustamov 2008] can transform intrinsic symmetries of a 2D shape into extrinsic ones in a high dimensional signature space, Ovsjanikov et al. [2008] presented a purely algebraic algorithm for detecting global intrinsic symmetries of shapes and classifying symmetries into discrete classes. Finally, the recent work of Bronstein et al. [2009] dealt with partial intrinsic similarity of non-rigid shapes and their technique can also be applied to detect different types of symmetries. However, it finds the largest part of a shape that is symmetric and in the case of Figure 2(b), it would identify the front-and-back symmetry which is unnatural.

3 IRSA: intrinsic reflectional symmetry axis

Given a closed 2-manifold \mathcal{M} embedded in \mathbb{R}^3 , we first compute a scalar field, the IRSA transform, over \mathcal{M} , which reveals the prominent PIRS axes of the shape. In theory, the IRSA transform should be a continuous scalar field. However to obtain a practical algorithm, we resort to a sampling-based voting approach (Section 3.3), which discretizes and approximates the problem. From the scalar field, we extract a set of explicit IRSA curves (Section 3.4) which capture the prominent PIRSeS of the shape represented by \mathcal{M} .

We start our coverage on this core section of the paper with some background (Section 3.1) and a review (Section 3.2) of the PRST work of Podolak et al. [2006], which has inspired the voting algorithm developed in our work. However, we point out several key challenges associated with PIRS analyses in the manifold setting.

3.1 Background

Let \mathcal{M} be a closed compact 2-manifold embedded in \mathbb{R}^3 , the 3D Euclidean space. Following Raviv et al. [2007], we say that \mathcal{M} possesses a global intrinsic symmetry if there is a homeomorphism $T : \mathcal{M} \rightarrow \mathcal{M}$ which is an isometry. That is, T preserves all the geodesic distances, denoted by $d_{\mathcal{M}}$. Thus for all $p, q \in \mathcal{M}$, $d_{\mathcal{M}}(p, q) = d_{\mathcal{M}}(T(p), T(q))$. Since we are interested in *partial* intrinsic symmetries, the homeomorphism T should be applied to a subset of \mathcal{M} . In our current work, we insist that the subset considered is connected. Furthermore, we are not interested in symmetries that are too local, thus we wish to find *maximal* partial intrinsic symmetries over \mathcal{M} . Specifically, a connected subset $\tilde{\mathcal{M}}$ of \mathcal{M} possess a maximal (partial) intrinsic symmetry if there is an isometric homeomorphism $\tilde{T} : \tilde{\mathcal{M}} \rightarrow \tilde{\mathcal{M}}$ and no connected subset of \mathcal{M} containing $\tilde{\mathcal{M}}$ possesses an intrinsic symmetry.

Intrinsic reflectional symmetry and reflection axis Formally, a reflection R is an *involute isometry* of an Euclidean space whose set of fixed points is an affine subspace of codimension 1. Moreover, the set of fixed points form the reflection axis. For R to be an involute, we require $R(R(p)) = p$ for all p in the domain of R . Generalizing the above concepts to the intrinsic setting only necessitates replacing the Euclidean distance by geodesic distance. On the 2-manifold \mathcal{M} , an intrinsic reflection axis should be a curve, which can generally assume an arbitrary shape; the curve may even be disconnected, e.g., consider the case where the shape is a torus.

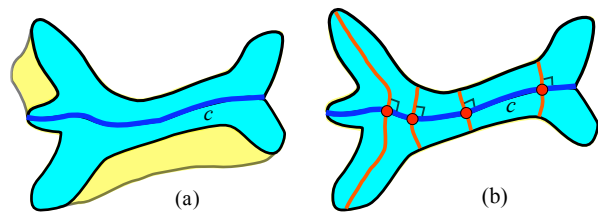


Figure 4: The symmetry generating set of the blue curve c is shown as the cyan region in (a), a subset of the whole shape. The local “thickness” of the generating set at an IRSA point (red dot) is given by the length of the orange geodesic segment (b) orthogonal to c at the point. Note that these figures are for illustration only.

Symmetry axes and generating sets For any given curve $c \subset \mathcal{M}$, we can define an intrinsic reflectional symmetry *generating set* $\mathcal{G}(c)$ of c as a subset of \mathcal{M} with the following properties:

- $\mathcal{G}(c)$ is *connected* and $c \subseteq \mathcal{G}(c)$;
- there is an intrinsic reflectional symmetry R over $\mathcal{G}(c)$ whose fixed point set is precisely c .

The maximal such $\mathcal{G}(c)$ is denoted by $\mathcal{G}^*(c)$. Obviously, not all curves have non-empty symmetry generating sets. If $\mathcal{G}(c)$ is non-empty, then c is an IRSA of $\mathcal{G}(c)$; see Figure 4(a).

Local reflectional symmetry support We can view the set $\mathcal{G}(c)$ as formed by the union of an infinite set of geodesic segments that are perpendicular to its IRSA c ; see Figure 4(b). For the maximal generating set $\mathcal{G}^*(c)$, each such geodesic segment would reach its maximal length without breaking the isometry constraint. These maximal segment lengths provide the “thickness” values for the generating set along the IRSA. One may also view these thickness values as indicating the *support of local reflectional symmetries* along the IRSA curve c .

IRSA transform To reveal the prominent IRSAs over \mathcal{M} , it is reasonable to accumulate over \mathcal{M} the “prominence” of each curve as an IRSA. In this paper, we simply model the prominence of a partial reflectional symmetry by the *area* of its defining domain. Specifically, given a curve $c \subset \mathcal{M}$, we can designate the area of $\mathcal{G}^*(c)$, the maximal reflectional symmetry generating set of c , as the *IRSA score* at c . Integrating the IRSA scores over all possible curves on \mathcal{M} , we obtain the continuous IRSA transform of \mathcal{M} .

Voting An algorithmic realization of the above integration would resort to sampling, letting a sample set of curves “vote” for their IRSA scores. However, the space of all possible curves over \mathcal{M} is too large. A more efficient alternative is to sample pairs of points from \mathcal{M} and let these pairs vote for their respective IRSA curves. Note that our thought process and analysis so far have resembled those from the PRST work of Podolak et al. [2006], which was developed in the planar extrinsic setting.

3.2 Review of planar reflective symmetry transform

The idea of planar reflective symmetry transform, or PRST, is intuitive and its computation via the voting paradigm is simple and elegant [Podolak et al. 2006]. Roughly speaking, given a shape f and a plane γ , a measure of f ’s extrinsic reflectional symmetry with respect to γ is the extent of overlap between f and its reflection about γ . Given a 2D shape whose boundary is specified by a closed contour, with a voting analogy, a plane γ having more point pairs on or near the boundary that are symmetric with respect to γ gets more votes and is a more prominent symmetry axis.

To practically compute PRST via voting, each pair of points p and q , with higher probability of choosing points on the boundary, vote for their reflection line, which is the bi-infinite boundary between the Voronoi cells of p and q in the plane. This implies a transform from the 2D shape to the space of lines, accounting for votes from all the pairs. The local maxima in the transform domain reveal reflection lines which correspond to prominent partial reflectional symmetries. In the original domain of the 2D shape, a scalar field which represents the accumulation of voted lines can be visualized and the prominent reflectional symmetry axes manifest themselves as darkened ridges. Our IRSA transform seeks to compute a similar scalar field, but over a manifold for PIRS.

Key challenges To adapt the voting paradigm for PRST to closed manifolds, we are met with several key challenges:

- In contrast to Euclidean lines or planes, the space of potential IRSAs consists of arbitrary curves on a manifold and does not admit a low-dimensional parameterization for binning.
- In the extrinsic view, a 2D or 3D shape is defined by its boundary which plays a key role in symmetry analysis. In particular, the boundary is heavily biased during Monte Carlo sampling when voting for PRST. The intrinsic view, on the other hand, while holding the key to achieving pose invariance in symmetry analysis, has no boundary to work with on the manifold surface. Thus a new bias or weighting scheme needs to be introduced into the voting strategy.
- Any reflection in the Euclidean space can be easily constructed, but the same cannot be said about an isometric “folding” within a closed manifold along a curved boundary. Distances, paths, and angles are all more expensive and less robust to deal with in the intrinsic setting.

3.3 Voting-based IRSA transform on manifold

The basic premise of the voting approach is that the weight of a vote from a point pair should correspond to the extent of PIRS possessed by that pair. A pair of points are highly symmetric if they have similar local geometry as well as large support from other symmetric point pairs, where two point pairs support each other if they share the same PIRS or IRSA. Such support is a more global criterion for symmetry analysis based on the isometry constraint.

Let us give a coarse-to-fine description of our voting algorithm. At a high level, for each *sampled, eligible* pair of points $a, b \in \mathcal{M}$, we let them vote for their reflectional symmetry boundary; see Algorithm 1. Unless otherwise specified, all the samplings are uniform. Here, eligibility is first provided by the *local shape similarity* between a and b . The reflectional symmetry boundary between a and b is the boundary $\beta_{\mathcal{M}}(a, b)$ between the intrinsic Voronoi cell of a and b : $\beta(a, b) = \{r \in \mathcal{M} : d_{\mathcal{M}}(a, r) = d_{\mathcal{M}}(b, r)\}$, where we ignore the reference to \mathcal{M} as it is clear from the context.

A key difference to PRST is that the symmetry boundary $\beta(a, b)$ must have *spatially varying* weights, as the filtering of point pairs using local shape similarity alone is insufficient to capture intrinsic reflectional symmetries over \mathcal{M} . These weights should indicate the strength or support of local intrinsic reflectional symmetry along $\beta(a, b)$, and as discussed in Section 3.1, they are precisely the thicknesses of the maximal symmetry generating set $\mathcal{G}^*(\beta(a, b))$.

One straightforward way to measure the local symmetry support is to trace geodesic segments from points along the boundary $\beta(a, b)$, orthogonally to $\beta(a, b)$, while ensuring isometry. However, such an approach is difficult to realize, expensive, and sensitive to geometric noise. We thus resort to a statistical approach relying on voting again. Essentially, we allow all the eligible sample point pairs (with eligibility determined by additional filters) to vote for the symmetry support at each point $m' \in \beta(a, b)$. We localize the

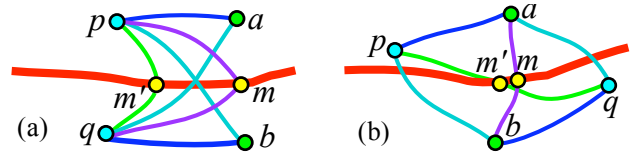


Figure 5: The distance filter test (color-matched curve segments are to have the same geodesic length) for $\{a, b\}$ and $\{p, q\}$ is passed (a), and at the same time, p, q are symmetric about $\beta(a, b)$, shown in red. However in (b), passing the test does not imply an agreed PIRS — p, q are not reflectionally symmetric about $\beta(a, b)$.

vote by considering the distance from m' to the voting points — larger distances imply weaker voting strengths. The weighted votes are determined by a Gaussian.

Pseudocode for the outer and inner voting procedures are given in Algorithms 1 and 2, respectively. We denote the IRSA transform by τ and the local symmetry support by μ . During voting for symmetry support, each pair $\{p, q\}$ must pass a series of filters. The obvious ones are related to local shape similarity and geodesic distances. However, these can be insufficient as we explain in our discussion on filtering. The four filters we implement are executed *in order*. The way we order them is based on their computational cost. Naturally, expensive checks are to be performed on a more selective set, i.e., they are applied in the later stages.

Similarity filter This filter ensures that the points a and b voting for a boundary should have sufficiently similar local neighborhoods geometrically. Any reasonable local shape descriptor can be applied to compare the neighborhoods. We use *SDF map*, an adaptation of the curvature map signature of Gatzke et al. [2005], replacing curvature measures by the shape diameter function (SDF) of Shapira et al. [2008]. Specifically, for a point p , we construct b equally spaced geodesic rings around it — two adjacent rings bound a geodesic bin. We compute SDF averages in each bin, resulting in

Algorithm 1 Voting for IRSA transform τ .

Given a set S of sample points on \mathcal{M} .

$\tau(x) = 0$ for all $x \in \mathcal{M}$

for $a, b \in S$ **do**

if **Similarity-Filter**(a, b) passed **then**

for $m' \in \beta(a, b)$ **do**

$\tau(m') = \tau(m') + \mathbf{SymmetrySupport}(a, b, m', S)$.

end for

end if

end for

Algorithm 2 Voting for local reflectional symmetry support μ .

$\mu(m') = \mathbf{SymmetrySupport}(a, b, m', S)$

$h = 0$.

for $p, q \in S$ **do**

 /* filters below executed *sequentially* */

if **Similarity-Filter**(p, q) passed **and**

Distance-Filter(p, q, a, b, m') passed **and**

Locality-Filter(p, q, a, b) passed **and**

Axial-Sym-Filter(p, q, a, b) passed **then**

$h = h + e^{-d_{\mathcal{M}}(p, m')^2 / \sigma^2}$

 /* Gaussian width $\sigma = \max_{x, y \in S} d_{\mathcal{M}}(x, y)$ */

end if

end for

return h .

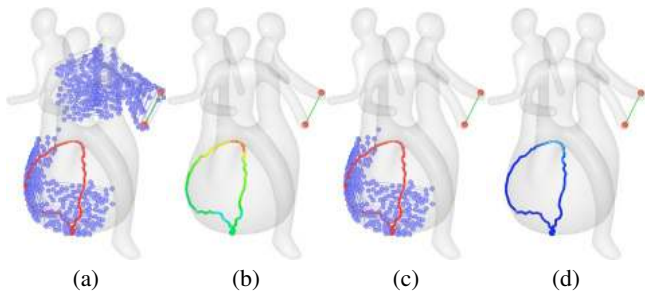


Figure 6: Effect of locality filtering on symmetry support voting. Without such filtering, the region spanned by the point pairs having passed previous filters can be disconnected (a). Here, the red dots and red curve are the voting point pairs $\{a, b\}$ and their Voronoi boundary $\beta(a, b)$, respectively. The symmetry support from voting (b) is erroneously large (red=large; blue=small). After density-based clustering, remaining eligible point pairs (c) have voted for the more proper symmetry support (d) for $\beta(a, b)$.

a b -dimensional signature, where we set $b = 6$ in all the examples. We choose SDF over conventional curvature estimates since SDF is more robust to noise and captures more global geometric information. The dissimilarity between two signatures is given by their L_2 distance. The similarity filter test is passed if the dissimilarity is less than ϵ_{SIM} of the maximum norm of the two compared signatures, where ϵ_{SIM} is a user-specified threshold parameter.

Distance filter This filter serves as the base-line test of isometry based on pairwise geodesic distances between the sample points involved. Given the two pairs $\{a, b\}$ and $\{p, q\}$ as specified above and in Algorithms 1 and 2, as well as the point m' along the tested boundary, we require that each pair of points lie on different sides of the boundary being voted on. Further, let m be the (geodesic) midpoint between a and b , we require via thresholding that $d_{\mathcal{M}}(a, p) = d_{\mathcal{M}}(b, q)$, $d_{\mathcal{M}}(a, q) = d_{\mathcal{M}}(b, p)$, $d_{\mathcal{M}}(p, m) = d_{\mathcal{M}}(q, m)$, and $d_{\mathcal{M}}(p, m') = d_{\mathcal{M}}(q, m')$; see Figure 5(a) for the configuration. The distance filter test is passed if each distance difference is less than ϵ_{DIS} of the maximum of the two compared distances, where ϵ_{DIS} is a user-specified threshold parameter.

Unfortunately, fulfilling the above distance checks does not guarantee that $\{a, b\}$ and $\{p, q\}$ will share the same PIRS, as illustrated in Figure 5(b). This problem is addressed by the final filter. Another problem to resolve is that point pairs which do not belong to the symmetry generating set of $\beta(a, b)$ may still participate in the voting so far. The locality filter described next tackles this issue.

Locality filter As the example in Figure 6(a) illustrates, given points a, b and their Voronoi boundary $\beta(a, b)$, the set of point pairs which could pass the distance filter may result in a disconnected set over \mathcal{M} . This indicates that the maximal symmetry generating set of $\beta(a, b)$ cannot be extended to include the whole point set, since over a region empty of detected point pairs, there cannot be an intrinsic reflectional symmetry. Thus we should enforce a *locality* criterion in pair selection: only point pairs local to $\beta(a, b)$ should vote. We implement this filter by finding the *single cluster* of points which contain $\beta(a, b)$. Only a point pair entirely belonging to this cluster is allowed to proceed to the next filter. We use an implementation of DBSCAN [Ester et al. 1996], which is a density-based clustering scheme, to identify the dense cluster of eligible point pairs close to $\beta(a, b)$; see Figure 6 for some results.

Axial-Sym filter Having identified the appropriate cluster of eligible voters and performed the similarity and distance tests, we wish now to filter out remaining point pairs $\{p, q\}$ that are not re-

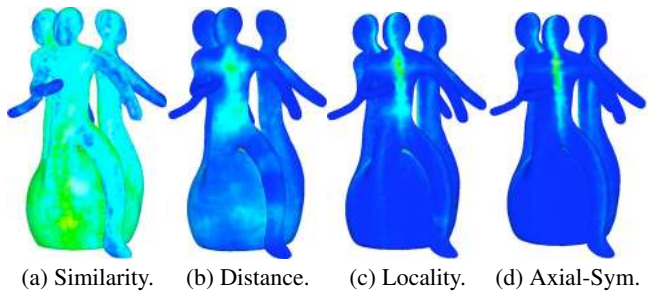


Figure 7: Effects of progressively applying more filters on the IRSA transform computed. (a)-(d): filters are added one at a time.

flexionally symmetric about $\beta(a, b)$. The existence of such point pairs can be seen from Figure 5(b), where we observe that the points a, b, p , and q roughly form a (geodesic) parallelogram. Such a configuration reveals an approximate isometry about a point — a *point symmetry*, as the set of fixed points of the implied isometric transformation contains a single point — the center of the parallelogram.

To evaluate the degree to which the pair $\{p, q\}$ support the reflectional symmetry, which is *axial*, about the boundary $\beta(a, b)$, we resort to a statistical approach again and call the filter the Axial-Sym filter. The idea is that if $\{p, q\}$ were reflectionally symmetric about $\beta(a, b)$ instead of having a point symmetry, then there would be more eligible point pairs (those having passed the previous filters) that agree to the symmetry $\{p, q\}$ induces. For efficiency considerations, we only apply a distance check. One could also utilize criteria involving geodesic angles. However, we have found that tracing geodesic paths to measure angles is expensive and noise-prone.

For the distance check within the Axial-Sym filter, we count the number of eligible point pairs $\{s, t\}$ which satisfy: $d_{\mathcal{M}}(s, p) = d_{\mathcal{M}}(t, q)$ and $d_{\mathcal{M}}(s, q) = d_{\mathcal{M}}(t, p)$; the same equality threshold ϵ_{DIS} as in the distance filter is used. All those pairs receiving a count below a user-specified percentage threshold ϵ_{AXS} are prevented from voting. The percentage is measured against the total number of eligible point pairs having passed the first three filters.

In Figure 7, we show the effects of the different filters on the obtained IRSA transform. However, it is important to note that despite the carefully designed filtering process, there is no theoretical guarantee that each remaining eligible voting pair $\{p, q\}$ share the same maximal PIRS over \mathcal{M} as the pair $\{a, b\}$. The robustness of our approach owes to the statistical nature of the voting scheme and the power of the data ensemble. More results demonstrating the quality of our voting scheme are given in Section 4.

3.4 IRSA curve extraction

Although the IRSA transform τ , the scalar field computed over \mathcal{M} via voting, reveals and accentuates the prominent IRSAs of the shape, the transform itself is not directly usable for subsequent shape analysis. What we desire is an explicit set of IRSA *curves* over \mathcal{M} which capture the prominent PIRSeS. In particular, each IRSA curve should induce a PIRS region (a symmetry generating set) and these PIRS regions constitute a *non-overlapping partition* of the shape into reflectionally symmetric patches. In this section, we describe an iterative scheme to obtain the set of IRSA curves.

To initialize, we select a top few Voronoi boundaries, based only on the current IRSA transform τ and symmetry support measures μ (estimated by Algorithm 2) along these boundaries. Potential issues with the initial set include: they may contain erroneous segments and they are unnecessarily constrained to be closed. These can be fixed by a process we call *IRSA pruning*, where we cut off or short-circuit segments of an IRSA curve with low symmetry support μ . To do so robustly however, we need more accurate μ values.

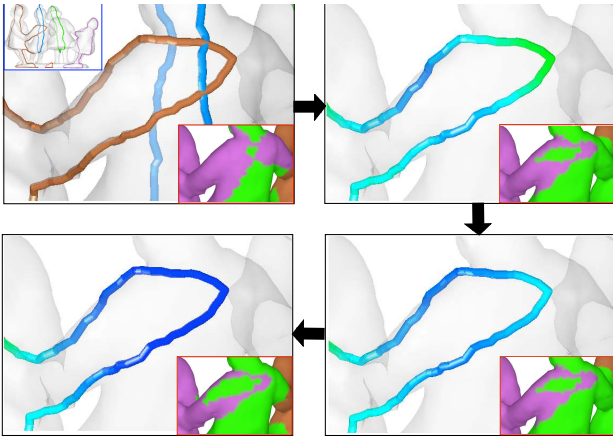


Figure 8: A few iterations of grass-fire region growing (regions colored purple in the bottom-right insert) and the resulting color plot of the local symmetry support measure μ (red=large, blue=small) along the IRSA curves, focusing on one particular IRSA curve. The extraneous segment has gradually lower support (blue) as iterations proceed and can be subsequently cut from the IRSA curve by IRSA pruning; see the final set of IRSA curves in Figure 1(c).

The key improvement offered by the symmetry support refinement algorithm in this section is to ensure that voting point pairs are *not shared* between different IRSAs. The locality filter presented in the previous section attempted to achieve a similar effect, but the above requirement was not enforced. We accomplish our goal by explicitly constructing regions associated with the IRSAs and by requiring that the voting pairs must come from the same region.

Initial IRSA curve extraction Among all Voronoi boundaries defined by the sample point pairs that passed the similarity filter, we select the top few serving as the initial IRSAs. Specifically, for each considered pair $\{a, b\}$, we compute the line integral

$$\nu(a, b) = \int_{\beta(a, b)} \tau(x) \cdot \mu(x) dx$$

and use it to rank the pair’s (closed) Voronoi boundary. The first boundary chosen is the one having the largest ν value. After having selected a Voronoi boundary, we set the IRSA transform value in the vicinity of that boundary to zero and then repeat. For a stopping criterion, we examine and look for a significant drop in the average ν values from one selected boundary to the next. If a drop is larger than the sum of the previous two or it exceeds a certain ratio r of the previous drop, then we stop. Here r is a user-specified parameter having a default value of $r = 0.8$.

Iterative symmetry support refinement During iterative refinement, we combine a symmetry-support-controlled region growing step (starting from IRSA curves) and symmetry support voting, where for the latter, the point pairs are confined by the obtained regions. Specifically, at each iteration, we perform a standard (simultaneous, isotropic) grass-fire region growing from each IRSA curve, with speed of growth determined by the symmetry support along the IRSA curves and the direction of growth orthogonal to the IRSA curves; see Figure 8 for a few results.

When the growth fronts all come to a stop, for each obtained region, we select only point pairs that lie within that region and also have passed all the four filters discussed in Section 3.3 to vote, as done in Algorithm 2, so as to recompute the symmetry support μ of the IRSA contained in the region. The iteration refinement stops when the symmetry support of all the IRSA curves have stabilized.

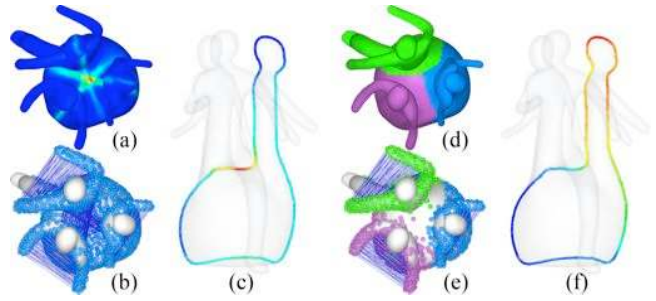


Figure 9: Region constraints on pair selection alleviate issues with rotational symmetry. (a) IRSA transform on the Momento model; observe a peak (red spot) due to approximate rotational symmetry among the three characters. (b) Original set of eligible (symmetric) pairs which voted for the IRSA transform are shared among different characters. (c) Color plot of initial symmetry support along an IRSA curve. (d) Regions obtained after grass-fire growing. (e) Eligible pairs grouped by regions eliminate point sharing. (f) Color plot of the new symmetry support along the IRSA curve.

Figure 8 shows an illustration of the evolution of the symmetry support values and the obtained regions.

An important advantage of our iterative approach lies in its handling of *rotational symmetries* in the shape. A rotational symmetry generally induces reflectional symmetries, where the different PIRS regions can overlap significantly. In other words, points from different PIRS regions can participate in the voting of different IRSAs. The region-constrained pair selection mechanism we follow addresses this issue, leading to symmetry support measures that more accurately capture the *exclusive* PRIS’s in the shape. See Figure 9 for an illustrative example.

IRSA pruning Up to now, all the obtained IRSA curves are still closed: they start as closed Voronoi boundaries and do not change spatially during refinement. However, as shown in Figure 9(f), this constraint is unnecessarily restrictive for an IRSA curve to derive the appropriate PIRS. Through IRSA pruning, we eliminate segments along an IRSA curve that have low symmetry support, e.g., the segment around the base of the Momento from Figure 9(f). In Figure 1(b-c), we see that the IRSA curve in yellow was partially removed. In our experiments, after linearly scaling the symmetry support values μ along an IRSA curve to $[0, 1]$, we remove any segment whose μ values fall below 0.25.

Removal of the low-support segments allows the remaining IRSA curve to produce a more meaningful PIRS region via region growing. However, under circumstances where one might want closed IRSA curves, we can simply replace the removed segment by a geodesic to close the gap. This is illustrated in Figure 1(c), where we see that one segment of the yellow IRSA curve (near the skirt) has been “short-circuited” by a geodesic segment. In the case of Figure 9(f) however, the geodesic connecting two ends of the blue segment does not have sufficient support, thus the gap left by removal of the blue segment remains open; the final set of IRSA curves for the Momento can be found in Figure 12.

3.5 Implementation

Our presentation so far has used a language applicable to the continuous setting. In reality, the input to our algorithm is a closed 2-manifold triangle mesh. Sample points are taken at mesh vertices, uniformly distributed, where we assume that the input mesh possesses a regular tessellation. If this were not the case, then a remeshing scheme could be applied. In Algorithms 1 and 2 and throughout Section 3.3, points $p, q, a,$ and b are always sample points, while the

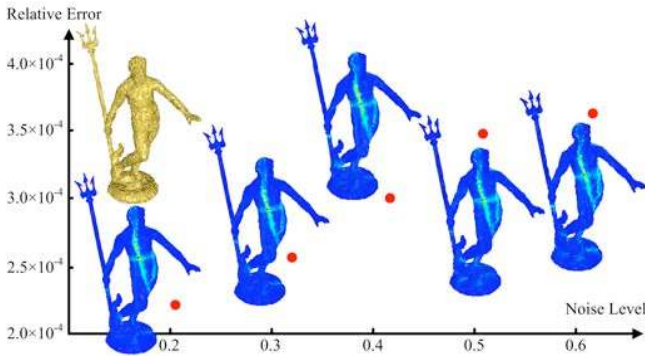


Figure 10: IRSA transform under synthetic Gaussian noise. The noise level is given as the average vertex displacement over the average edge length in the original mesh. The top-left image shows the noisy model at level 0.42. Discrepancy between IRSA transforms is given by the relative error: $\|\hat{\tau} - \tau_0\|_2 / \|\tau_0\|_2$, where τ_0 and $\hat{\tau}$ are the IRSA transform vectors for the original and noisy mesh models, respectively, and $\|\cdot\|_2$ is the L_2 norm.

voted boundaries $\beta(a, b)$ contain only mesh vertices; in particular, both m and m' are mesh vertices, but not necessarily in the sample set S . In all of our experiments, the mesh models have between 10K to 15K vertices (the original dense models were decimated) and the number of samples chosen range between 2K and 5K. Our geodesic computation uses Danil Kirsanov’s implementation of the algorithm of Surazhsky et al. [2005].

Note also that our description of the algorithm and the pseudocode given are presented in a way for better conceptual clarity. The actual implementations have considered standard optimization for efficiency. As a simple example, the density-based clustering algorithm is executed only once per point pair $\{a, b\}$ in Algorithm 2 instead of within every call to clustering-based thresholding, as it would appear in the pseudocode of Algorithm 2.

We rely on four filters to select point pairs during voting for the IRSA transform, where the user needs to select the appropriate thresholds. These selections have been carried out experimentally and once an appropriate parameter set have been chosen, we adhere to them. All the IRSA transforms shown in the paper have been produced using the same threshold settings. Specifically, $\epsilon_{SIM} = 0.025$, $\epsilon_{DIS} = 0.02$, and $\epsilon_{AXS} = 80\%$.

4 Results and applications

In this section, we first demonstrate the robustness of our voting scheme in the presence of moderate noise and moderate amount of missing data such as small holes. In the first test, we progressively add synthetic Gaussian noise to perturb the mesh vertex positions and then apply the voting scheme to compute the IRSA transform τ , using exactly the same parameter settings. However the set of sample points may vary due to random uniform sampling. Figure 10 is representative and it shows both visually (by comparing with the result in Figure 12) and quantitatively (by looking at the relative error) that our voting scheme is quite robust against the added noise.

Figure 11 shows several IRSA transforms computed on models with artificially inserted small holes. Locations and sizes of the holes were randomly selected but under some control. Indeed, due to its reliance on geodesic distances, our algorithm is not designed to handle significant missing data such as those arising from raw scans. Appropriate preprocessing steps for data cleaning and repair should be applied under those circumstances. However, by comparing with results given in Figures 1 and 12 on the same set of models, the results in Figure 11 demonstrate that our voting scheme can well tolerate small amount of missing data.

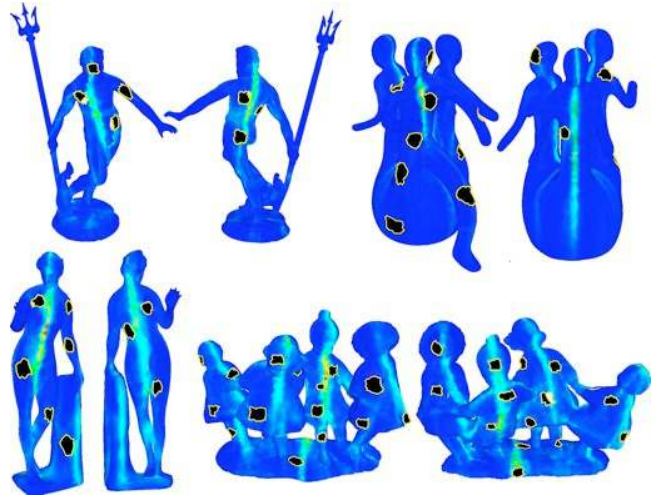


Figure 11: IRSA transform in the presence of small holes: Neptune (7 holes, with 2.32% triangles missing), Momento (13, 2.93%); the sculpture woman (9, 2.25%), and the children model (27, 3.87%). Two views per example are provided to show the inserted holes. The relative errors (defined in the same way as for Figure 10) measured against the original complete models are: 3.6×10^{-4} , 6.3×10^{-4} , 2.7×10^{-4} and 9.6×10^{-4} , in that order.

Figure 12 shows more results for IRSA transform and IRSA curve extraction. Next, we investigate how the IRSA curves can be used for two applications which benefit from PIRS analysis.

4.1 Symmetry-aware mesh segmentation

Symmetry-aware mesh segmentation builds upon the premise that a shape part is typically symmetric. It is an attractive and powerful cue which captures shape semantics. Yet it requires no prior knowledge to detect — purely geometric schemes suffice.

With the ability to extract prominent PIRSeS of 3D shapes, we are able to identify highly complex, intrinsically reflectionally symmetric parts from a shape; see the gallery in Figure 12. From Section 3.4, we already have a symmetry-aware segmentation scheme via grass-fire region growing from the final set of IRSA curves. This approach would be the intrinsic analogue of an extrinsic symmetry-aware segmentation scheme, e.g., [Podolak et al. 2006; Simari et al. 2006]. A result using this approach is shown in the top-left corner of the gallery, the “snowmen”, demonstrating that this scheme is capable of finding cut boundaries over entirely smooth surface regions. Such a result is purely induced by symmetry and not attainable using existing algorithms which searches for concavity. However, this toy example is too perfect. Most models do not possess perfect PIRSeS, thus purely relying on region growing based on symmetry support measured along a set of given IRSAs will inevitably introduce less than meaningful segmentation boundaries.

A variety of options are possible, including post-processing of boundaries and incorporating feature weights into region growing. We have experimented with the idea of using IRSA-induced PIRS regions to perform merging of segments produced by a black-box mesh segmentation algorithm, e.g., anyone of those which utilizes the minima rule [Shamir 2006]. The key is to *over-segment* the model using the black-box algorithm. Then we assign each segment to the PIRS region which provides the largest coverage (in area) among all PIRS regions from the symmetry-induced grass-fire region growing. Figure 12 shows five results obtained and they demonstrate the robustness of our voting and IRSA curve extraction schemes, as well as our ability to obtain highly semantic mesh segmentations compared to conventional approaches.

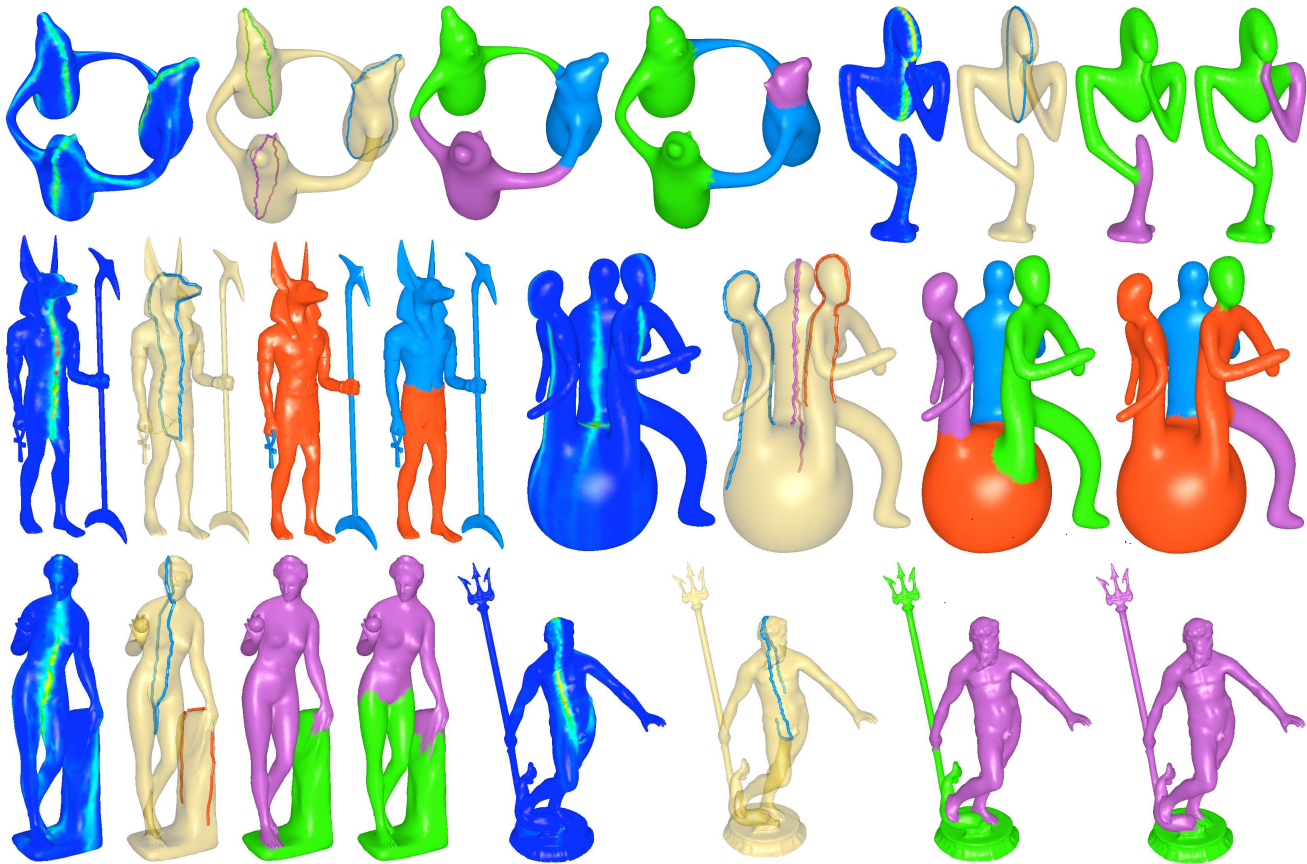


Figure 12: A gallery of results for IRSA transform, IRSA curve extraction, and symmetry-aware mesh segmentation. For each model, the first three images show the above results, in that order. The last image shows the result of using the segmentation algorithm of Liu and Zhang [2007], with the same number of parts as for its symmetry-aware counterpart. For the snowmen model, purely symmetry-driven region growing was used. For the rest, symmetry-based merging was performed on an over-segmented set of parts from [Liu and Zhang 2007].

We experimented with several state-of-the-art mesh segmentation schemes. For each, we over-segment an input model into 20 parts. Choosing the proper number of IRSA curves, which becomes the target number of parts for merging, is carried out via thresholding as described in Section 3.4. With the exception of the children model in Figure 1, all the examples shown in the paper were produced using the default parameter setting. Although segmentation results from different choices of the black-box algorithms vary, the symmetry-aware results have been consistent throughout.

4.2 Symmetry-driven part repair

Our second application is symmetry-driven part repair. Given an intrinsically reflectionally symmetric shape with certain parts missing, we wish to repair it by performing PIRS analysis on the incomplete shape, which may be asymmetric or whose global intrinsic symmetry is an unnatural one, and recovering the missing parts and restoring the shape’s symmetry. Our investigation into this problem is still preliminary and we aim to demonstrate its potential.

The basic idea is to first perform the grass-fire region growing from an IRSA curve to obtain an intrinsically reflectionally symmetric region. Over the region, the symmetric point pairs induce a point correspondence. Discrepancies at the region boundaries allow us to detect the missing parts. See Figure 13(a-b) for an illustration on the David model, where the right arm is regarded as an asymmetric part of this model and the region boundary symmetric to the base of that arm is where the missing part will be inserted.

Owing to symmetry recovery, the above correspondence implies the part instance to use for the repair. We make a copy of the part instance and position it by performing a rigid alignment between the corresponding region boundaries. For the David model in Figure 13, these are the bases of the two arms. However, it is difficult to determine the initial positioning of the part instance that would lead to the most natural part repair. Here we resort to a heuristic which generally works well for human-like shapes. Namely, we reflect the part copy with respect to the best-fitting plane of the IRSA curve, and then perform rigid alignment using the standard iterated closest point (ICP) algorithm. A few ICP iterations produce a conforming overlap region between the two boundaries. We then use Poisson interpolation [Yu et al. 2004] to stitch the parts and obtain a single manifold surface; see Figure 13(d). Figure 14 shows two more examples of symmetry-driven part repair.

It is worth pointing out the importance of our PIRS analysis during the process, which detects the proper symmetry axis for a shape part. With an arm missing, a human-like shape still possesses an approximate global intrinsic symmetry, the one between the front and the back of the body. It is conceivable that such a symmetry would be detected by a method designed to recover a global intrinsic symmetry, but it would not have led to a proper part repair.

5 Conclusion and future work

While there have been algorithms designed to detect a variety of global or partial extrinsic symmetries and global intrinsic symme-

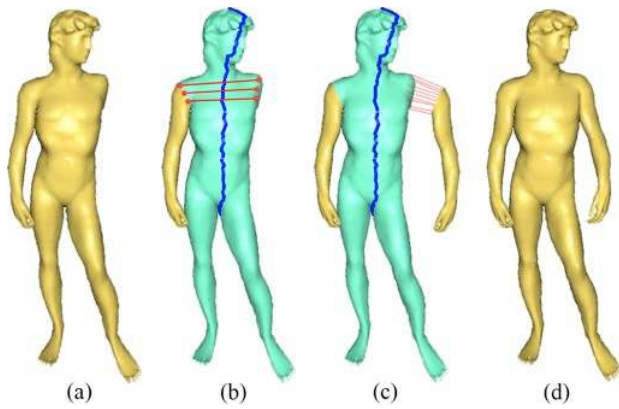


Figure 13: Part repair on the David model. (a) Initial model with the left arm missing. (b) Detected PIRS region (green) and point correspondences (red lines). (c) Initial point correspondence between the symmetric region boundaries. (d) Rigid alignment of the left arm with the body and stitching using Poisson interpolation.

tries in 2D or 3D shapes, we are not aware of any work capable of detecting *partial* intrinsic reflectional symmetries over a closed 2-manifold. The detection of structures in subsets, implying a much larger search space (e.g., comparing partial matching to full correspondence), and the common challenges one must face when working in the intrinsic surface setting, make the problem at hand difficult. Inspired by early works on symmetry detection, in particular [Podolak et al. 2006], we develop a voting scheme to compute an intrinsic reflectional symmetry axis (IRSA) transform of a 3D shape. Our scheme is robust due to its statistical nature and with the aid of a grass-fire region growing method and an iterative refinement procedure, we obtain quality IRSA curves which can be subsequently utilized in symmetry-aware mesh segmentation and symmetry-driven part repair. The use of symmetry in these applications adds useful semantic information to the mix.

Limitations Our current algorithm and implementation still leave much room for improvement. First, our work is constrained to detecting reflectional symmetries only. The study of partial intrinsic rotational or translational symmetries may call for a different approach. Secondly, the voting approach developed dictates that only PIRSeS having sufficiently large support can be reliably detected. In cases where semantically meaningful PIRSeS are present on relatively small parts of a complex model, our current voting scheme may fail. Third, the reliance on geodesic distances could render our approach incapable of handling large amount of noise, missing data, and certain high-genus models, though we have demonstrated certain level of robustness and shown positive results on several high-genus models — see Figures 10-12. Finally, a major limitation of our current method is speed. The worst-case time complexity of our method is $O(n^4)$, where n is the number of sample points — for each point pair that votes for a Voronoi boundary, we examine the set of all point pairs to compute the local symmetry support values. The series of filters alleviate the complexity, however, currently, on a mesh with 10K vertices and with 3K samples, the voting process takes about 2 hours on an Intel Core 2 Duo 6300, 1.86 GHz machine with 1GB memory. The time taken for IRSA curve extraction and for the two applications are comparatively negligible, as we typically work only on a small number of IRSA curves.

Future work In addition to finding more efficient filtering mechanisms and resorting to multi-resolution or parallelization techniques (e.g., using the GPU) to improve the speed of the voting algorithm, we would also like to explore new extensions and applications of

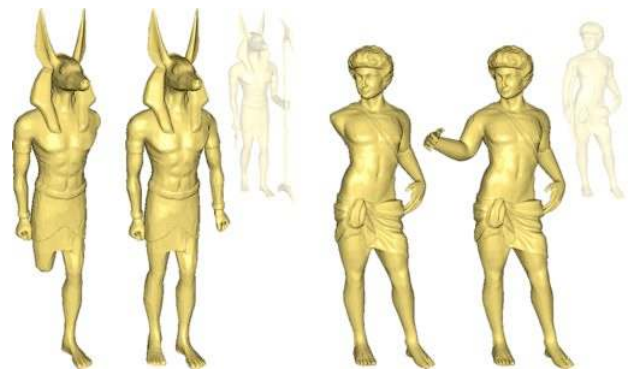


Figure 14: More examples of symmetry-driven part repair. Left: with two missing parts. Right: over a shape with imperfect symmetry. The original models are shown at the upper-right corners.

PIRS analysis. We believe that its potential has not been fully realized. One possible extension is to construct a hierarchical shape decomposition based on PIRSeS. Symmetry-induced shape correspondence analysis also seems to be a direction worth exploring. Finally, while the use of IRSA curves in the mesh segmentation framework has led to semantically meaningful results compared to state-of-the-art algorithms, we have only utilized information provided by the IRSA curves in a simplistic manner. It is desirable to more intelligently combine the symmetry criterion with conventional measures applied to solve the mesh segmentation problem, where an optimization-based framework may be appropriate.

Acknowledgments

The authors are grateful to the reviewers for their valuable comments. The mesh segmentation code was provided by Rong Liu and the SDF code was originally developed by Lior Shapira. Thanks also go to Patricio Simari, Karan Singh, Ariel Shamir, and Daniel Cohen-Or for fruitful discussions. The Momento model was courtesy of Oscar Au and the remaining models were either from the AIM@SHAPE Shape Repository or captured by our Polhemus laser scanner. This work is supported in part by grants from NSERC (No. 611370), the 973 National Key Basic Research Foundation of China (2009CB320801), the 863 Program of China (No. 2007AA01Z313), NSFC (No. 60773022), the China Scholarship Council and the joint grant of the National Natural Science Foundation of China and Microsoft Research Asia (60776799).

References

- ATALLAH, M. J. 1985. On symmetry detection. *IEEE Trans. Comput.* 34, 7, 663–666.
- BOKELOH, M., BERNER, A., WAND, M., SEIDEL, H.-P., AND SCHILLING, A. 2009. Symmetry detection using line features. *Computer Graphics Forum (Special Issue of Eurographics)* 28, 2, 697–706.
- BRONSTEIN, A. M., BRONSTEIN, M. M., AND KIMMEL, R. 2006. Generalized multidimensional scaling: A framework for isometry-invariant partial surface matching. *Proc. National Academy of Sciences (PNAS)* 103, 5, 1168–1172.
- BRONSTEIN, A. M., BRONSTEIN, M., R., AND KIMMEL. 2007. Calculus of non-rigid surfaces for geometry and texture manipulation. *IEEE Trans. Vis. & Comp. Graphics* 13, 5, 902–913.
- BRONSTEIN, A. M., BRONSTEIN, M. M., BRUCKSTEIN, A. M., AND KIMMEL, R. 2009. Partial similarity of objects, or how to compare a centaur to a horse. *Int. J. Comp. Vis.* 84, 2, 163–183.

- CHAOUCH, M., AND VERRROUST-BLONDET, A. 2008. A novel method for alignment of 3D models. *Proc. IEEE Int. Conf. on Shape Modeling and Applications*, 187–195.
- CHEN, X., GOLOVINSKIY, A., AND FUNKHOUSER, T. 2009. A benchmark for 3D mesh segmentation. *ACM Trans. on Graph* 28, 3, 73:1–73:12.
- ELAD, A., AND KIMMEL, R. 2001. Bending invariant representation for surfaces. *Proc. IEEE Conf. on Comp. Vis. and Pat. Rec.*, 168–175.
- ESTER, M., KRIEGEL, H.-P., SANDER, J., AND XU, X. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. *Proc. of Int. Conf. on Knowledge Discovery and Data Mining*, 226–231.
- GAL, R., AND COHEN-OR, D. 2006. Salient geometric features for partial shape matching and similarity. *ACM Trans. on Graph* 25, 1, 130–150.
- GATZKE, T., GRIMM, C., GARLAND, M., AND ZELINKA, S. 2005. Curvature maps for local shape comparison. *Proc. IEEE Int. Conf. on Shape Modeling and Applications*, 244–256.
- GOLOVINSKIY, A., PODOLAK, J., AND FUNKHOUSER, T. 2007. Symmetry-aware mesh processing. *Princeton University TR-782-07*.
- HOFFMAN, D. D., AND RICHARDS, W. A. 1984. Parts of recognition. *Cognition* 18, 65–96.
- HOFFMAN, D. D., AND SINGH, M. 1997. Saliency of visual parts. *Cognition* 63, 1, 29–78.
- KAZHDAN, M., CHAZELLE, B., DOBKIN, D., FINKELSTEIN, A., AND FUNKHOUSER, T. 2002. A reflective symmetry descriptor. *Proc. Euro. Conf. on Comp. Vis.* 2, 642–656.
- KAZHDAN, M., CHAZELLE, B., DOBKIN, D., FUNKHOUSER, T., AND RUSINKIEWICZ, S. 2004. A reflective symmetry descriptor for 3D models. *Algorithmica* 38, 1, 201–225.
- KAZHDAN, M., FUNKHOUSER, T., AND RUSINKIEWICZ, S. 2004. Symmetry descriptors and 3D shape matching. *Symp. on Geom. Proc.*, 115–123.
- KÖHLER, W. 1929. *Gestalt Psychology*. Liveright, New York.
- LEYTON, M. 1992. *Symmetry, Causality, Mind*. MIT Press.
- LEYTON, M. 2001. *A Generative Theory of Shape*. Lecture Notes in Computer Science, Vol. 2145. Springer.
- LIU, R., AND ZHANG, H. 2007. Mesh segmentation via spectral embedding and contour analysis. *Computer Graphics Forum (Special Issue of Eurographics)* 26, 3, 385–394.
- LOY, G., AND EKLUNDH, J.-O. 2006. Detecting symmetry and symmetric constellations of features. In *Proc. Euro. Conf. on Comp. Vis.*, 508–521.
- MARTINET, A., SOLER, C., HOLZSCHUCH, N., AND SILLION, F. X. 2006. Accurate detection of symmetries in 3D shapes. *ACM Trans. on Graph* 25, 2, 439–464.
- MITRA, N. J., GUIBAS, L. J., AND PAULY, M. 2006. Partial and approximate symmetry detection for 3D geometry. *ACM Trans. on Graph* 25, 3, 560–568.
- MITRA, N. J., GUIBAS, L. J., AND PAULY, M. 2007. Symmetrization. *ACM Trans. on Graph* 26, 3, 63:1–63:8.
- OVSJANIKOV, M., SUN, J., AND GUIBAS, L. 2008. Global intrinsic symmetries of shapes. *Computer Graphics Forum (Proc. of Symposium on Geometry Processing)* 27, 5, 1341–1348.
- PAULY, M., MITRA, N. J., WALLNER, J., POTTMANN, H., AND GUIBAS, L. 2008. Discovering structural regularity in 3D geometry. *ACM Trans. on Graph* 27, 3, 43:1–43:11.
- PODOLAK, J., SHILANE, P., GOLOVINSKIY, A., RUSINKIEWICZ, S., AND FUNKHOUSER, T. 2006. A planar-reflective symmetry transform for 3D shapes. *ACM Trans. on Graph* 25, 3, 549–559.
- PODOLAK, J., GOLOVINSKIY, A., AND RUSINKIEWICZ, S. 2007. Symmetry-enhanced remeshing of surfaces. *Symp. on Geom. Proc.*, 235–242.
- RAVIV, D., BRONSTEIN, A. M., BRONSTEIN, M. M., AND KIMMEL, R. 2007. Symmetries of non-rigid shapes. *Proc. Int. Conf. on Comp. Vis.*
- RIKLIN-RAVIV, T., KIRYATI, N., AND SOCHEN, N. 2006. Segmentation by level sets and symmetry. *Proc. IEEE Conf. on Comp. Vis. and Pat. Rec. 1*, 1015–1022.
- RIKLIN-RAVIV, T., KIRYATI, N., AND SOCHEN, N. 2007. Laplace-beltrami eigenfunctions for deformation invariant shape representation. *Symp. on Geom. Proc.*, 225–233.
- RUSTAMOV, R. 2008. Augmented planar reflective symmetry transform. *The Visual Computer* 24, 6, 423–433.
- SHAMIR, A. 2006. Segmentation and shape extraction of 3D boundary meshes. *Eurographics STAR Report*, 137–149.
- SHAPIRA, L., SHAMIR, A., AND COHEN-OR, D. 2008. Consistent mesh partitioning and skeletonization using the shape diameter function. *The Visual Computer* 24, 4, 249–259.
- SIMARI, P., KALOGERAKIS, E., AND SINGH, K. 2006. Folding meshes: hierarchical mesh segmentation based on planar symmetry. *Symp. on Geom. Proc.*, 111–119.
- STEWART, I., AND GOLUBITSKY, M. 1992. *Fearful Symmetry: Is God a Geometer?* Blackwell Cambridge, MA.
- SURAZHISKY, V., SURAZHISKY, T., KIRSANOV, D., GORTLER, S. J., AND HOPPE, H. 2005. Fast exact and approximate geodesics on meshes. *ACM Trans. on Graph* 24, 3, 553–560.
- THRUN, S., AND WEGBREIT, B. 2005. Shape from symmetry. *Proc. Int. Conf. on Comp. Vis.*, 1824–1831.
- WEYL, H. 1983. *Symmetry*. Princeton University Press.
- WOLTER, J. D., WOO, T. C., AND VOLZ, R. A. 1985. Optimal algorithms for symmetry detection in two and three dimensions. *The Visual Computer* 1, 1, 37–48.
- YEH, Y., AND MECH, R. 2009. Detecting symmetries and curvilinear arrangements in vector art. *Computer Graphics Forum (Special Issue of Eurographics)* 28, 2, 707–716.
- YU, Y., ZHOU, K., XU, D., SHI, X., BAO, H., GUO, B., AND SHUM, H.-Y. 2004. Mesh editing with poisson-based gradient field manipulation. *ACM Trans. on Graph* 23, 3, 644–651.
- ZABRODSKY, H., AND WEINSHALL, D. 1997. Using bilateral symmetry to improve 3D reconstruction from image sequences. *Computer Vision and Image Understanding* 67, 48–57.