Folding meshes: Hierarchical mesh segmentation based on planar symmetry

Patricio Simari, Evangelos Kalogerakis and Karan Singh[†]

DGP Lab, Department of Computer Science, University of Toronto

Abstract

Meshes representing real world objects, both artist-created and scanned, contain a high level of redundancy due to (possibly approximate) planar reflection symmetries, either global or localized to different subregions. An algorithm is presented for detecting such symmetries and segmenting the mesh into the symmetric and remaining regions. The method, inspired by techniques in Computer Vision, has foundations in robust statistics and is resilient to structured outliers which are present in the form of the non symmetric regions of the data. Also introduced is an application of the method: the folding tree *data structure. The structure encodes the non redundant regions of the original mesh as well as the reflection planes and is created by the recursive application of the detection method. This structure can then be* unfolded *to recover the original shape. Applications include mesh compression, repair, skeletal extraction of objects of known symmetry as well as mesh processing acceleration by limiting computation to non redundant regions and propagation of results.*

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computational Geometry and Object Modeling]: Geometric algorithms, languages, and systems. Curve, surface, solid, and object representations. Hierarchy and geometric transformations.

1. Introduction

Symmetry plays a fundamental role in nature, manifested both in the form and function of living organisms. Visually, symmetry is important to humans, as it influences our perceptual understanding of objects in the world. Symmetric patterns, not surprisingly, are an important design principle in guiding the aesthetic and construction of synthetic objects [Arn54, Gom69]. Neuroscience research goes so far as to indicate that aspects of symmetry in humans may be hardwired into our visual processing system [NCP*02]. Symmetries are ubiquitous in humans, our environment and our creations of art and architecture.

The classification, understanding and intelligent representation of shape is an active area of research in geometry processing. Recognizing the common presence of symmetries in many real world objects can greatly assist in solutions

to various shape representation problems such as simplification, repair, noise removal and skeletal extraction. Of the various types of symmetries found, planar symmetry is perhaps the most commonplace and is thus the focus of this paper. While planar symmetries have been recognized to be an important feature in shape understanding, there has been little work in shape representations that are defined as a structured assembly of symmetric parts. In this paper we present the concept of a *folding tree* (see figure 4), where an object is defined as a hierarchical union of planar symmetric and asymmetric parts. Each nested detection of a symmetric part reduces the complexity of representation of said part by half, greatly simplifying the overall representation complexity of many objects.

For folding trees to be useful beyond an academic concept, we must be able to automatically construct them from geometric data such as meshes as well as regenerate the object from its folding tree. Central to folding tree construction is the problem of automatically finding a maximally symmetric part of the object. We observe that most organic ob-

---

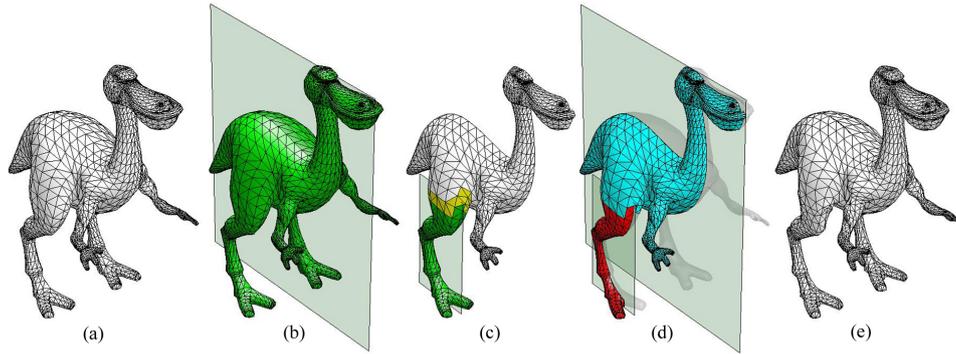[†] {psimari, kalo, karan}@dgp.toronto.edu

Figure 1: *(a) Original dino mesh. (b) Detected global symmetry plane. (c) Remaining half dino mesh with a detected local symmetry. Green indicates the region of symmetry and yellow indicates faces still in the support region but not included in the symmetric region. (d) Remaining leaf geometry (the two leaves shaded a different color). (e) Dino reconstructed from the leaf geometry and symmetry planes.*

jects with planar symmetry are articulated figures with coherent symmetric parts that are connected at the joints of an underlying skeletal structure. Most synthetic objects show a construction history involving symmetric primitives, symmetry creating operations such as reflections, planar symmetry preserving operations like revolves and coherent combinations of various symmetric parts often with some asymmetric refinement. Motivated by these observations we additionally constrain our problem to finding a maximal symmetric part that is a single connected surface component of the object. The constraint has several advantages, including simplifying reconstruction. Multiple surface components with the same planar symmetry are just as easily represented as multiple symmetry nodes that have the same symmetry plane.

Overview: Our approach to finding maximally symmetric parts is based on robust M-estimation using an iteratively reweighted least squares (IRLS) algorithm. We simultaneously solve for the reflection plane as well as the region of surface that is symmetric with respect to it. Upon convergence, the symmetric region found is separated from the rest. The algorithm is then applied to the remaining regions to find other local symmetries, as well as recursively to half of the symmetric region to find nested symmetries. This process leads to a hierarchical folding tree representation of the object where geometry is only stored in tree leaves. The original surface can then be reconstructed from its folding tree in a bottom up fashion by reflecting symmetric geometry and reconnecting segmentation boundaries.

Contribution: This paper presents two principal contributions. Firstly, we introduce a method capable of detecting *global* as well as *local* approximate planar symmetries in 3D meshes. Our algorithm has solid foundations on statistical methods for robust M-estimation [HRRS86] which has also been used in many recent computer vision applica-

tions [SAG95, BA96, Ste99]. Secondly, we exploit our symmetry detection approach for *mesh folding*: the elimination of planar symmetry redundancy from the mesh data. Our algorithm is orthogonal to other existing methods for mesh compression [Hop96, GBTS99, KG00, IA02] which do not explicitly take advantage of repeating symmetric areas in 3D meshes. (For a recent survey, see [AG05].)

The rest of the paper is organized as follows: section 2 describes the related work on symmetry detection and applications; section 3 describes our method for detecting global and local symmetries on mesh data; section 4 introduces folding trees and describes their construction; section 5 shows our results; and finally, section 6 presents our conclusions and future work.

2. Related Work

Although the computation of symmetries in shapes has been an intriguing area of research in computer vision and computational geometry literature for the last 30 years at least, to our knowledge, there has been comparatively little research on symmetry detection in 3D meshes mainly due to the increased complexity of the existing algorithms when extended from the 2D to the 3D case. Our work differs from most approaches for 3D symmetry detection in meshes in that we aim at the robust detection of not only global but also local reflection symmetries, i.e. those present only in parts of a 3D mesh, and we exploit these symmetries in order to achieve mesh compression by eliminating faces implied by the discovered symmetries.

The detection of symmetries in 2D and 3D models has mainly been applied in object classification, recognition and reconstruction. Early approaches dealt with symmetries of planar point sets by applying pattern recognition algorithms that search for matches in circular strings represent-

ing the graphs of polyhedral objects [Ata85,WWV85,Hig86, JYB96].

Despite the optimality of these algorithms that could also detect all the possible symmetries in a shape, they were only able to recover perfect symmetries in 2D and 3D shapes making them useless in the presence of small perturbations, imprecision and noise which is very common in meshes. The problem of approximate symmetries was addressed in [AMWW88] that considered the problem of computing generic geometric transformations between two point sets. The paper gives a detailed theoretical analysis of the developed algorithm for symmetry, however, it deals with global symmetry and it is unclear if the given algorithm could be extended to three dimensions. Such is also the case of other methods for finding symmetries of symmetric or almost symmetric 2D planar images [Mar89, GK96, SICT99].

An interesting extension of that early work which introduced the notion of symmetry distance, meaning how much of a given symmetry an object possesses, was developed in [ZPA95]. The approach can evaluate symmetries in the presence of noise and also find locally symmetric regions in 2D images and 3D range data. The reflection plane of the image is determined by minimizing the symmetry value over all possible reflection planes using a gradient descent algorithm to locate the plane of maximal symmetry. More recently, an approach also taking 3D range images as input was presented in [TW05]. A probabilistic measurement model is used to detect symmetries in order to reconstruct partially occluded 3D shape models. Although both methods can find local symmetries, they follow a greedy technique searching in growing localized regions. In contrast, our method is top down, leading to the gradual removal of asymmetric outliers from the region for the robust detection of the maximal symmetric area using an M-estimator approach. In this regard, our approach tends to find the largest areas of symmetry first, avoiding over-segmentation.

Another original approach that detects the dominant hyperplane of bilateral symmetry in range images of 3D objects with a linear time algorithm is presented in [MO96]. The hyperplane is uniquely defined by the centroid and eigenvectors of the covariance matrix of the object. This method is limited to the detection of the plane of global symmetry and is not robust to outliers or imprecision in the 3D object. Sun *et. al* [SS97, SS99] address the symmetry detection problem by, in the first case, searching for correlations in the 3D object's gaussian image, and in the second, by using the image's orientation histogram in a similar fashion. More recently, Martinet *et al.* [MSHS05] recover symmetries by examining the extrema and spherical harmonic coefficients of the object's generalized moments. These approaches, however, also focus strictly on global symmetry detection.

Symmetry shape descriptors are introduced in [KCD*04, KFR04] where a collection of spherical functions are used to describe the measure of rotational and reflective sym-
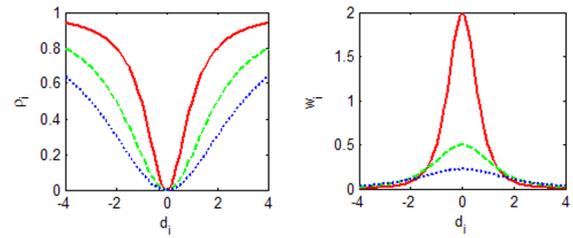

Figure 2: *GM estimator cost* ρ *(left) and associated weight* w *(right) as a function of distance for scale parameter* $\sigma = 1$ *(red solid),* $\sigma = 2$ *(green dashed) and* $\sigma = 3$ *(blue dotted).*

metry present in a mesh with respect to every axis passing through its center of mass. The descriptors had several desirable properties such as robustness and stability. However, the approach aimed at using global symmetry information as a shape descriptor and not at extracting local symmetries.

In parallel with our work, two very interesting symmetry detection algorithms have been developed. In the first, Podolak *et al.* propose the planar reflective symmetry transform (PRST) [PSG*06] as a shape descriptor. For any given plane, the PRST indicates the degree of symmetry which the object exhibits with respect to it. For efficiency of computation, the authors propose a Monte Carlo framework, in which pairs of randomly selected points vote for the plane between them. These votes are accumulated in discrete bins over polar parameters in a manner reminiscent of the Hough transform, and local maxima are later refined. These local maxima correspond to planes for which the object exhibits a degree of local or global symmetry. In the second, Mitra *et al.* [MGP06] also propose to consider pairs of points and their determined symmetry. Rotational, and translational symmetries are also considered, as well as scaling. Instead of a bin counting scheme, representative symmetries are extracted from the transform space through mean shift clustering. Both proposed methods seem robust and efficient. On the other hand, our approach represents an easy-to-implement alternative symmetry detection method which is also based on robust statistics along with a hierarchical simplification scheme that can be directly applied to mesh compression and reconstruction, in both cases through our folding tree structure.

3. Symmetric region detection

Given a mesh, we wish to find a connected region S of faces that exhibit planar symmetry within a tolerance parameter ε. In the case of global symmetry, this region should be the entire mesh. We approach the problem as a model fitting scenario, in which the model consists of the sought plane, and the connected region of symmetry.

Distance metric: given a plane p, we denote the distance

from a point r_i^p, which is the reflection of v_i with respect to p, to a mesh M as $d_i^p = dist(r_i^p, M)$. For notational simplicity we will avoid including the p superscript in the following.

We compute the distance function *dist* from a reflected vertex to the mesh by taking the minimum point-to-triangle distance from the point to the closest compatible face on mesh M [RL01]. We consider a face to be compatible with a given query vertex if the angle between the interpolated normal at the closest point on the face and the vertex's reflected normal is less than 45 degrees.

Given the presence of structured outliers in the form of the non symmetric regions of the mesh, we interleave solving for the symmetric region S and the plane p based on an iteratively reweighted least squares (IRLS) approach, using the Geman-McClure (GM) robust M-estimator [HRRS86, FP02]. In essence, the GM estimator maps error values to an associated cost. This cost approaches constant (and the associated weight approaches zero) as error values approach infinity, thus mitigating the influence of outliers on the minimization process. The GM estimator exhibits excellent behavior in rejecting structured outliers with the appropriate choice of the scale factor σ [SAG95]. This parameter essentially controls the rate with which weight decreases as error increases. (See figure 2.)

Solving for the plane: Given the current distances d_i, the GM cost estimator ρ_i and associated weight w_i for each vertex are given by

$$\rho_i = \frac{d_i^2}{\sigma^2 + d_i^2} \qquad w_i = \frac{1}{d_i}\frac{\partial \rho_i}{\partial d_i} = \frac{2\sigma^2}{(\sigma^2 + d_i^2)^2}$$

In addition, in order to be robust in the presence of tessellations with varying face sizes, we multiply the obtained weights by their associated vertex areas, i.e. $w_i \leftarrow w_i \frac{1}{3}\sum_{j=1}^{k} area(f_j)$, where $f_1,...,f_k$ are the faces incident on vertex v_i.

For a body which exhibits planar symmetry it is known that its plane of symmetry is perpendicular to a principal axis and contains the object's center of mass [MIK92]. This lets us solve for the current plane of maximum symmetry in a closed form manner by considering the center of mass m and weighted covariance matrix C relative to the weights w_i.

$$m = \frac{1}{s}\sum_{i=1}^{n} w_i v_i \qquad C = \frac{1}{s}\sum_{i=1}^{n} w_i(v_i - m)(v_i - m)^T$$

where $s = \sum_i w_i$.

We compute the eigenvectors of C and consider the three planes determined by these vectors and m. For each of these planes we compute the distances d_i and associated costs ρ_i retaining the one of minimum sum cost. This now lets us solve for the support region.

Support region: Given the current ρ values and a candidate face $f = (v_1, v_2, v_3)$ we consider it to be a *support* face if it holds that $\forall_{i \in \{1,2,3\}} d_i \leq 2\sigma$ [HRRS86]. We then find the
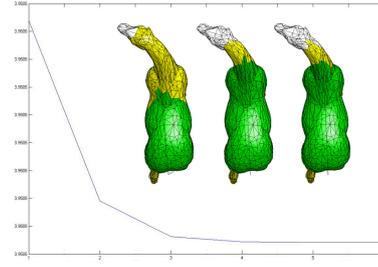

Figure 3: *Illustration of algorithm convergence. The plot shows the objective function $\sum_i \rho_i$ for support vertices v_i. The placement of the estimate of the symmetry plane along with support region (yellow) and symmetric region (green) are shown for the horse model. Left to right, iterations 1, 5 and 10 respectively, using the precise distance metric.*

largest connected region of support faces, taking this as our new estimate of the support region and set the weights for all vertices outside this region to be 0, thus controlling leverage.

The estimation and region finding steps are iterated until convergence.

Initialization and details: Initially we simply define w_i to be the mesh area associated with vertex v_i as defined above, and the initial support region contains all faces.

In order to accelerate and improve convergence we initially use a discrete approximation of the above distance function. For a given vertex v_i and face f_j this distance function $dist_{discrete}(v_i, f_j)$ is defined as the Euclidean distance from the reflected vertex r_i to the face plane of f_j if the angle between the reflected normal and f_j is less than 90 degrees, and infinite otherwise. The distance $dist_{discrete}(v_i, M)$ is defined as the distance to the f_j whose centroid is closest to v_i. During these initial iterations we set $\sigma = 1.4826 \cdot median(d_i)$, which is a popular estimate of scale [FP02], not letting it fall below 2ε to avoid instability.

This discrete distance function is first used until convergence, after which the more precise distance function *dist* is used. At this point we set $\sigma = 2\varepsilon$. This setting allows for near-symmetric vertices to be included in the support region albeit with lower weight.

Upon convergence, the symmetric region S is extracted as the largest connected region of faces whose vertex distance values are all below ε. We detect convergence by comparing the current plane estimate with that of the previous iteration checking for a sufficiently small difference.

Convergence: In our experiments both distance functions exhibit very good convergence behavior (to their respective minima). Figure 3 illustrates an example of the convergence properties of our approach.

4. Folding trees

4.1. Definitions

We consider a region R of a mesh M to be a connected subset of the faces of M.

A segmentation $\{R_1, R_2, ..., R_n\}$ of a mesh M is a set of mutually exclusive regions whose union results in M.

A folding tree T representing a mesh M is inductively defined as one of the following:

- a *leaf* node, which contains mesh data for M.
- a *folding* node, with one subtree S, and a plane of symmetry p.
- a *segmentation* node, with n subtrees $T_1, T_2, ..., T_n$, where T_i is a folding tree for region R_i such that regions $R_1, R_2, ..., R_n$ are a segmentation of the mesh represented by T.

The *unfold* operation can now be defined on a folding tree T as follows.

- The mesh data M if T is a leaf.
- $unfold(S) \cup reflect(unfold(S), p)$, where S is the unique subtree of T, p is the reflection plane,
- $\cup_{i=1}^{n} unfold(T_i)$ where $T_1, T_2, ..., T_n$ are the subtrees of T.

Here, *reflect* indicates the mesh resulting from planar reflection of the argument mesh's vertices with respect to the argument plane.

4.2. Folding tree construction and unfolding

Given a mesh M, a folding tree T that represents it can be constructed in preorder through repeated application of the segmentation method of section 3. First, we apply the segmentation algorithm to M to find a subregion of planar symmetry, S, also obtaining the plane of symmetry p. We remove S from M and consider the set of remaining connected components $\{R_1, R_2, ..., R_n\}$. Note that in the case of a global symmetry, this set will be empty. We now construct a folding tree T with $n+1$ children $T_0, T_1, ..., T_n$, each representing $S, R_1, R_2, ..., R_n$ respectively. We know S to be symmetric with respect to plane p, so we can now *fold* S, retaining half of its surface S'. In particular, T_0 will be a folding node, labeled with p, and its child T_0' will represent S'. The resulting structure is illustrated in figure 4. The subtrees $T_0', T_1, ..., T_n$ can now be created recursively with regions $S', R_1, ..., R_n$ respectively as inputs.

When discarding half of the faces of a particular region, it must be decided which half to keep, which to discard, and which to modify, if any. Because of varying tessellation and the provided tolerance, both sides need not be identical. In our implementation we keep the side with the most faces in order to preserve detail. Alternatively, we could keep the half with the least faces in order to minimize storage. Faces with all vertices on the discarded side of the plane are removed. In addition, faces that straddle the symmetry plane are clipped.
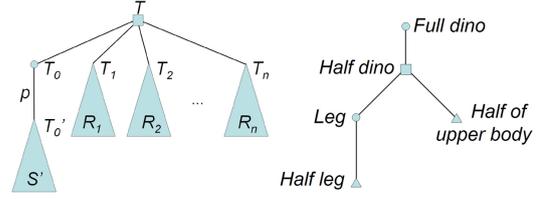
Figure 4: *Left: Folding tree construction structure when a local symmetry is detected. Right: Example of a complete folding tree for the dino mesh decomposition of figure 1. Circles represent folding nodes, squares segmentation nodes, small triangles are leaf geometry and large triangles represent (recursive) folding trees.*
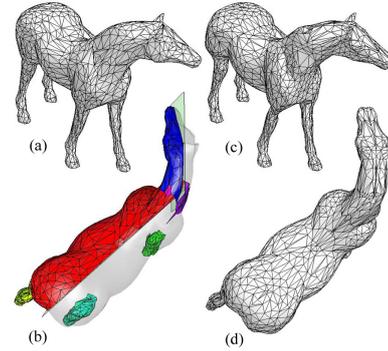

Figure 5: *(a) Original horse model. (b) Resulting tree leaves after folding and symmetry planes. Note that the local symmetries of the body and the articulated head were detected. (c) and (d) The reconstructed horse model.*

The recursive construction of the tree may be stopped, triggering the creation of a leaf node, by using one or more criteria: when the number of faces in the mesh is below a given threshold, when the area of the mesh is below a certain percentage of that of the original mesh, or when the number of recursive folds exceeds a given maximum. Our implementation allows for any or all three.

The procedure for unfolding a tree consists of a postorder traversal according to the definition of subsection 4.1. Upon reconstruction, because of the tolerance parameter of the region-finding algorithm, as well as differences in tessellation, the resulting mesh may have vertex misalignments. These can be corrected automatically using known techniques [GTLH98, TL94, Ju04] or software. The unfolding drives this repair, the union of the definition of subsection 4.1 between unfolded parts indicating that these should be processed in this manner. Figures 1, 4 and 5 illustrate the concepts.
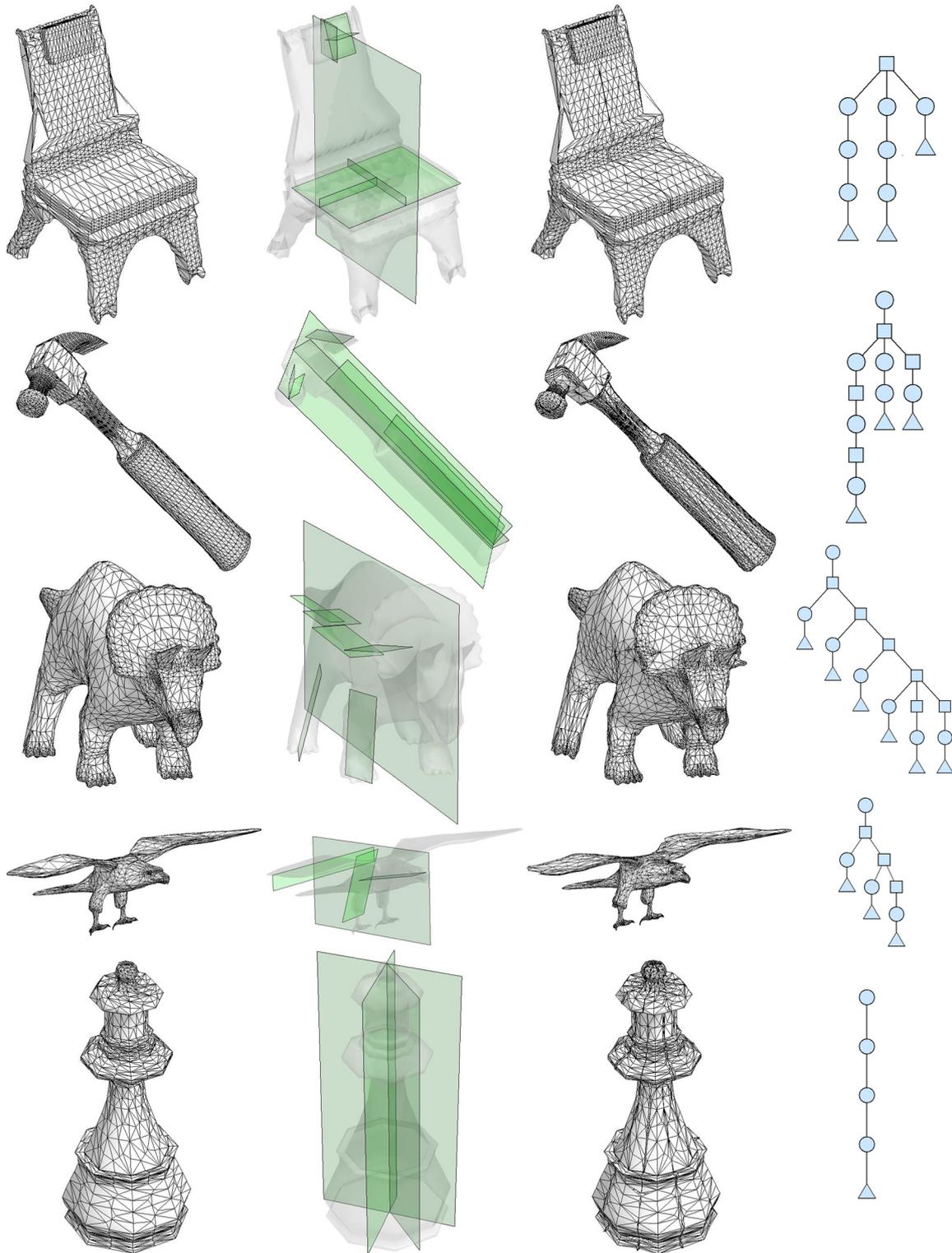
Figure 6: *1st column: original mesh data. 2nd column: All folding planes present in the constructed tree. 3rd column: model resulting from unfolding of folding-tree representation. 4th column: Folding tree. Circles represent folding nodes, squares segmentation nodes, and triangles are leaf geometry. Leaf nodes corresponding to asymmetric regions are not shown for clarity.*

Mesh	# Orig. f's	#Nonred. f's	Mesh bytes	Tree bytes	Comp.	Tree height	Rec. error	Time
Dino	6638	3142	265864	130916	50.76%	3	0.0358%	152 sec
Horse	3306	2672	120312	93932	21.93%	3	0.0004%	24 sec
Chair	5736	2460	229880	104352	54.61%	4	0.1600%	58 sec
Hammer	4360	677	174712	53120	69.60%	7	0.1764%	174 sec
Triceratops	5660	2447	226712	123696	45.44%	7	0.0147%	202 sec
Eagle	33072	15808	160440	96516	39.84%	5	0.0067%	936 sec
Queen	3360	600	134712	31952	76.28%	4	0.0001%	120 sec

Table 1: *Results for seven characteristic meshes. Columns from left to right: mesh name, number of faces in the original mesh, number of non redundant faces stored in folding tree leaves, original mesh storage in bytes, tree storage in bytes, compression achieved according to one minus the ratio of the previous two columns, height of the folding tree, reconstruction error as a percentage of the bounding box diagonal and running time, as reported by our Matlab 7 implementation.*

5. Results

The implementation of the symmetry detection algorithm and the folding tree representation of meshes, as described in sections 3 and 4, has been developed in Matlab 7. The user defines the tolerance parameter of the algorithm and the criteria for stopping the hierarchical segmentation of the mesh. The default tolerance value is 2% of the bounding box diagonal of the mesh. The default criterium for terminating the recursion is that the area of the current region is less than 5% of the total mesh area.

We present characteristic results, concerning mesh compression, the depth of the hierarchical segmentation, average reconstruction error as a percentage of the bounding box diagonal, as well as running times in table 1. Figure 6 shows initial and reconstructed meshes, complementing the results of figures 1 and 5, as well as illustrating all folding plane positions and the folding trees. In the chair model, we find the vertical plane of global symmetry then each cushion, which was a separate connected component, was folded through three perpendicular planes. The hammer is firstly folded in half through a vertical global plane of symmetry. Then, the handle is divided twice more. The cylindrical portion of the head is also subdivided twice more recursively. In the case of the triceratops and eagle models we find a global plane of symmetry and then local symmetries in the legs, tail and body for the triceratops model, and in the wings and upper legs of the eagle. Finally, in the case of the octagonal queen chess piece, all planar symmetries are recursively discovered resulting in one eighth of the original surface being stored.

Figure 7 further illustrates the symmetry detection approach. In the woman, we firstly find the dominant partial symmetry that includes her body and legs. Searching for nested symmetry, the algorithm detects the symmetry of the leg. Proceeding to the remaining regions, we find the local symmetries of the head and arms. Analogously, in the dragon, we find the symmetry of its body and then legs, head and arms. In the bull, we detect the local symmetry of its body including the back left leg, then symmetries in the other three legs and the head and finally another weak nested symmetry found in the middle of its body. We detect symmetries on the body, and separately in head and the ears of the bunny and also find two other weak nested symmetries in its body. Lastly, in the octopus, its head is found to be symmetric, also containing a nested symmetry, and multiple local symmetries are found in different parts of its tentacles. We would like to note that all meshes with the exception of the chair are originally a single connected component.

Our tests were run on an Intel Pentium M 2.13GHz processor under Matlab 7.

6. Conclusions and future work

We have proposed a novel approach for finding global planar symmetries in 3D meshes based on robust M-estimation. In addition, we have presented a new compact representation of meshes, called folding trees, which represent the original mesh by only encoding the non redundant regions as well as the planes of symmetry and can be used to recover the original object through unfolding.

Given the fact that real objects, both organic and synthetic, often exhibit this type of data redundancy and human perception is strongly related to the notion of symmetry, a significant number of applications based on our methodology can further be developed. The elimination of faces which are repeated in redundant areas of global and local symmetries leads to new mesh compression schemes that can be used for mesh storage, processing, and transmission. Automatic segmentation, reconstruction and repairing of the meshes, driven by the extracted symmetries, is also another interesting field of application of our method. The folding tree representation could also facilitate skeleton extraction and advanced editing operations which preserve symmetries.

Our future research will be focused on both the development of such applications as well as the exploitation of other types of symmetries in 3D meshes that can open up new implementations and extensions of our proposed methodology.

In cases where there is no strong symmetry in the neighborhood of the principal axes, our approach may fail to find
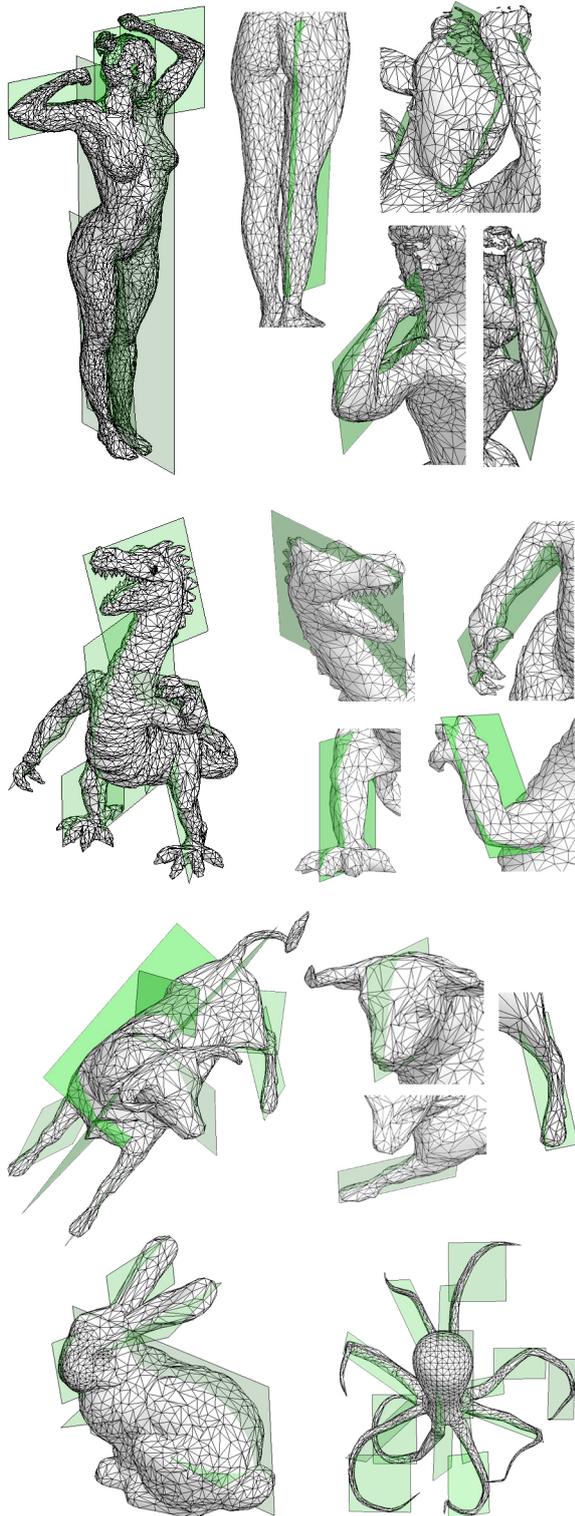
Figure 7: *Local symmetry planes identified by the method.*

existing local symmetries in small regions (for example, in the hands of the dino mesh of figure 1). In cases such as these in which our initial guess does not provide a large enough support, an alternative initialization can be sought through random sampling or perhaps through a voting scheme like the ones described by Mitra *et al.* [MGP06] or Podolak *et al.* [PSG*06].

References

[AG05] ALLIEZ P., GOTSMAN C.: Recent advances in compression of 3D meshes. In *Advances in Multiresolution for Geometric Modelling* (2005), Dodgson N., Floater M., Sabin M., (Eds.), Springer-Verlag, pp. 3–26.

[AMWW88] ALT H., MEHLHORN K., WAGENER H., WELZL E.: Congruence, similarity, and symmetries of geometric objects. *Discrete Compututational Geometry 3*, 3 (1988), 237–256.

[Arn54] ARNHEIM R.: *Art and Visual Perception: A Psychology of the Creative Eye*. University of California Press, Berkeley, 1954.

[Ata85] ATALLAH M.: On symmetry detection. *IEEE Trans. Computers 34*, 7 (1985), 663–666.

[BA96] BLACK M. J., ANANDAN P.: The robust estimation of multiple motions: parametric and piecewise-smooth flow fields. *Computer Vision and Image Understanding 63*, 1 (1996), 75–104.

[FKS*04] FUNKHOUSER T., KAZHDAN M., SHILANE P., MIN P., KIEFER W., TAL A., RUSINKIEWICZ S., DOBKIN D.: Modeling by example. *ACM Trans. Graph. 23*, 3 (2004), 652–663.

[FP02] FORSYTH D. A., PONCE J.: *Computer Vision: A Modern Approach*, first ed. Prentice Hall, 2002.

[GBTS99] GUEZIEC A., BOSSEN F., TAUBIN G., SILVA C.: Efficient compression of non-manifold polygonal meshes. In *Proceedings of the 10th IEEE Visualization 1999 Conference (VIS '99)* (Washington, USA, 1999), IEEE Computer Society.

[GK96] GOFMAN Y., KIRYATI N.: Detecting symmetry in grey level images: The global optimization approach. In *Proceedings of the 13th International Conference on Pattern Recognition* (1996), pp. 951–956.

[Gom69] GOMBRICH E. H.: *Art and Illusion: A Study in the Psychology of Pictorial Representation*. Princeton University Press, Princeton, 1969.

[GTLH98] GUEZIEC A., TAUBIN G., LAZARUS F., HORN W.: Converting sets of polygons to manifold surfaces by cutting and stitching. In *VIS '98: Proceedings of the conference on Visualization '98* (Los Alamitos, CA, USA, 1998), IEEE Computer Society Press, pp. 383–390.

[Hig86] HIGHNAM P. T.: Optimal algorithms for finding the symmetries of a planar point set. *Inf. Process. Lett. 22*, 5 (1986), 219–222.

[Hop96] HOPPE H.: Progressive meshes. In *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques* (New York, USA, 1996), ACM Press, pp. 99–108.

[HRRS86] HAMPEL F. R., RONCHETTI E. M., ROUSSEEUW P. J., STAHEL W. A.: *Robust Statistics: The Approach Based on Influence Functions.* Wiley-Interscience, New York, USA, 1986.

[IA02] ISENBURG M., ALLIEZ P.: Compressing polygon mesh geometry with parallelogram prediction. In *Proceedings of the conference on Visualization '02* (Washington, USA, 2002), IEEE Computer Society, pp. 141–146.

[Ju04] JU T.: Robust repair of polygonal models. *ACM Trans. Graph. 23*, 3 (2004), 888–895.

[JYB96] JIANG X., YU K., BUNKE H.: Detection of rotational and involutional symmetries and congruity of polyhedra. *Visual Computing 12*, 4 (1996), 193–201.

[KCD*04] KAZHDAN M., CHAZELLE B., DOBKIN D., FUNKHOUSER T., RUSINKIEWICZ S.: A reflective symmetry descriptor for 3d models. *Algorithmica 38*, 1 (2004), 201–225.

[KFR04] KAZHDAN M., FUNKHOUSER T., RUSINKIEWICZ S.: Symmetry descriptors and 3d shape matching. In *SGP '04: Proceedings of the 2004 Eurographics/ACM SIGGRAPH symposium on Geometry processing* (New York, NY, USA, 2004), ACM Press, pp. 115–123.

[KG00] KARNI Z., GOTSMAN C.: Spectral compression of mesh geometry. In *SIGGRAPH '00: Proceedings of the 27th annual conference on computer graphics and interactive techniques* (New York, USA, 2000), ACM Press/Addison-Wesley Publishing Co., pp. 279–286.

[Mar89] MAROLA G.: On the detection of the axes of symmetry of symmetric and almost symmetric planar images. *IEEE Trans. Pattern Anal. Mach. Intell. 11*, 1 (1989), 104–108.

[MGP06] MITRA N. J., GUIBAS L. J., PAULY M.: Partial and approximate symmetry detection for 3d geometry. *To appear in SIGGRAPH '06* (2006).

[MIK92] MINOVIC P., ISHIKAWA S., KATO K.: *Three Dimensional Symmetry Identification, Part I: Theory.* Tech. Rep. 21, Kyushu Institute of Technology, Japan, 1992.

[MO96] MARA D. O., OWENS R.: Measuring bilateral symmetry in digital images. In *In Proceedings of IEEE TENCON - Digital Signal Processing Applications* (1996), vol. 1, pp. 151–156.

[MSHS05] MARTINET A., SOLER C., HOLZSCHUCH N., SILLION F.: *Accurately Detecting Symmetries of 3D Shapes.* Tech. Rep. RR-5692, INRIA, September 2005.

[NCP*02] NORCIA A., CANDY R., PETTET M., VILDAVSKI V., TYLER C.: Temporal dynamics of the human response to symmetry. *Journal of Vision 2*, 2 (3 2002), 132–139.

[PSG*06] PODOLAK J., SHILANE P., GOLOVINSKIY A., RUSINKIEWICZ S., FUNKHOUSER T.: A planar-reflective symmetry transform for 3d shapes. *To appear in SIGGRAPH '06* (2006).

[RL01] RUSINKIEWICZ S., LEVOY M.: Efficient variants of the icp algorithm. *In proceedings of the Third International Conference on 3-D Digital Imaging and Modeling* (2001), 145.

[SAG95] SAWHNEY H. S., AYER S., GORKANI M.: Model-based 2d&3d dominant motion estimation for mosaicing and video representation. In *Proceedings of the Fifth International Conference on Computer Vision* (Washington, USA, 1995), IEEE Computer Society, p. 583.

[SICT99] SHEN D., IP H. S., CHEUNG K. T., TEOH E. K.: Symmetry detection by generalized complex moments: A close-form solution. *IEEE Trans. Pattern Anal. Mach. Intell. 21*, 5 (1999), 466–476.

[SS97] SUN C., SHERRAH J.: 3d symmetry detection using the extended gaussian image. *IEEE Trans. Pattern Anal. Mach. Intell. 19*, 2 (1997), 164–168.

[SS99] SUN C., SI D.: Fast reflectional symmetry detection using orientation histograms. *Real-Time Imaging 5*, 1 (1999), 63–74.

[Ste99] STEWART C. V.: Robust parameter estimation in computer vision. *SIAM Rev. 41*, 3 (1999), 513–537.

[TL94] TURK. G., LEVOY M.: Zippered polygon meshes from range images. In *SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1994), ACM Press, pp. 311–318.

[TW05] THRUN S., WEGBREIT B.: Shape from symmetry. In *Tenth International Conference on Computer Vision* (2005), vol. 2, IEEE Computer Society, pp. 1824–1831.

[WWV85] WOLTER J. D., WOO T. C., VOLZ R. A.: Optimal algorithms for symmetry detection in two and three dimensions. *Visual Compututer 1* (1985), 37–48.

[ZPA95] ZABRODSKY H., PELEG S., AVNIR D.: Symmetry as a continuous feature. *IEEE Trans. Pattern Anal. Mach. Intell. 17* (1995), 1154–1166.