

# Hybrid Hierarchical Shape Representation for Medical Shapes

Abhishek Kolagunda

Department of Computer and  
Information Science  
University of Delaware  
DE, USA

Guoyu Lu

Chandra Kambhamettu

<https://www.eecis.udel.edu/wiki/vims/>

---

## Abstract

Advances in 3D medical imaging technology have led to an increase of interest in shape analysis of organs. This has in turn led to explosion of 3D medical shape data being collected. 3D shape data is also being used for simulations and to guide minimally invasive and remote surgical procedures. We present a Hierarchical-Hybrid Shape Representation (HSSR) which is compact and has both explicit and implicit forms. The compactness of the representation largely reduces storage requirement and communication overheads. The explicit and implicit forms can be used for accurate visualization of organ shapes and guide surgical procedures. The hybrid shape model proposed is a combination of Extended Superquadrics and Radial Basis interpolation function (RBF) that separately models the base shape and surface deformations. We also present an automatic method to fit the hybrid shape model to complex shapes by hierarchically dividing the shape into parts. Finally, we propose a technique to reconstruct shape from its compact representation by recursively blending the parts using intersection shapes. Our extensive experiments show that our shape representation method significantly outperforms other existing approaches in both accuracy and compactness.

## 1 Introduction

Medical imaging techniques such as Computed Tomography (CT) and Magnetic Resonance (MR) play an important role at all levels of health-care system today. There has been increased interest in shape analysis, disease classifiers and other statistical analyses of the medical data which can be used to improve modern health-care system. This has led to huge amounts of high quality medical data being collected. Much effort is spent on storing, managing and sharing the data with collaborators [22, 23]. Recently, shape analysis has become of increasing interest in the medical community due to its potential in capturing the morphological variations across a population. The high quality 3D images captured can be used to extract 3D shape of the organs [19, 21]. 3D models of organs can also be used for training personnel, for visualization during image guided interventions and in simulations. The large amounts of 3D shape data generated by these systems pose challenges in storing, retrieving and visualizing 3D structures.

In minimally invasive surgical procedures, to help guide the surgeon, the 3D shape of the organ is captured prior to the procedure and is used for visualizing the organ and the deformation it undergoes during the procedure. Recent advances in MRI and robotic technologies have led to the development of MRI guided robotic interventions and remote surgical procedures [1, 2, 3, 4, 5]. A shape model that has implicit and explicit forms, which can model rigid and non-rigid deformations will assist in planning and monitoring trajectory of surgical instrument with respect to the organ surface. A compact representation for shape reduces storage requirement and communication delays for remote surgical procedures where the shapes have to be sent over a network for visualization at remote location.

Point Distribution Model [6] has been widely used to model 3D shapes of organs, mainly for performing statistical analysis of the shape and for use in deformable model based segmentation. NURBS are piece-wise parametric functions that are extensively used to model complex shapes. RBF interpolation functions have been widely used in the field of computer graphics to model 3D shapes as implicit surfaces [7, 8]. RBF and NURBS model shape as piece-wise local surface patches. The local shape representations while are accurate requires large number of parameters.

In [9] deformable superquadrics were used to model 3D objects. The authors used global deformations like tapering, bending and cavity. They used LM method to perform non linear least square fit. Hyperquadrics was introduced in [10] which models shape as combination of linear terms raised to powers. Zhou et al. [11] proposed the extended superquadric which can model a wide range of non-symmetric objects by replacing the constant exponents with functions. They used Bernstein polynomial (Bezier curves) for the exponent functions. These global representations with their compact form can model only limited set of shapes. There have been works where the global and local representations have been combined to compactly model more complex shapes. Bardin et al. [12] use deformable superquadrics introduced in [13], to model the global shape and then model the local deformations using free form deformations; they find control points on a grid to deform the initial superquadric fit. Tutar et al. [14] modeled the global shape using superquadric and the local deformations using Fourier descriptors to generate closed tubular surfaces. Spherical harmonics were used in [15] to model shapes of prostate. Lee et al. [16] represented a shape as a combination of low-resolution mesh and a deformation field. Though these representation can model relatively complex shapes they do not have both implicit and explicit forms. Hybrid Hyperquadrics was proposed in [17] which allowed both local and global control of shape. Extended superquadrics with cubic spline as exponent function were used in [18]. To model shape of complex 3D objects, authors iteratively segmented the object into multiple pieces across planes having error of fit greater than a threshold. They used genetic algorithm to solve for the extended superquadric parameters. Bischoff et al. [19] proposed a compact representation of the shape which is a combination of 3D points and ellipsoids. Chevalier et al. [20] modeled parts of the object using superquadrics. Parts based modeling of shapes are usually limited by the capabilities of the geometric functions used to model each part. Using a simple shape model for the parts affects the accuracy of fit and a complex shape model makes the fitting process computationally challenging.

We propose a compact hybrid shape model as a combination of Extended Superquadrics (ESQ) and RBF interpolation function. The two functions are combined to have both explicit and implicit forms. We also present a method to fit the shape model to complex shapes by hierarchically dividing the shape into approximately convex parts. To reconstruct the shape from its representation, we propose a method to blend parts using the intersection shape between parts.

## 2 Shape Representation

We use a binary tree representation for shape. Each node except the leaves contain three pieces of information: links to the left and right nodes representing constituent parts of the shape, and information required to combine its two constituent parts. A leaf node holds geometric representation of a part. We call this Hierarchical-Hybrid shape representation since a shape is hierarchically decomposed into its constituent parts and the parts are represented by a hybrid geometric model. Details of the hybrid geometric representation, the hierarchical decomposition of the shape and the method for combining decomposed parts are presented in the following sections.

### 2.1 Shape Model

Given shape is formed by a base shape and surface deformations. The hybrid shape model described here tries to capture these two components of the shape separately. ESQ is used to capture the base shape and RBF surface is used to capture the surface deformations. The two models are combined such that their combination has both explicit and implicit forms. The 3D shape model in its parametric form is given as  $f(\theta, \phi) = h(\theta, \phi) + g(\theta, \phi)$ .  $f$  is an injective function that maps points from the 2 dimensional space (Latitude and Longitude) to the 3 dimensional Euclidean space (X, Y, Z).  $h$  is the extended superquadric function and  $g$  is radial basis interpolation function. The points on the surface of the shape are given by

$$(X, Y, Z) = f(\theta, \phi)$$

$$\begin{aligned} X &= a \cdot \text{sign}(\cos\theta\cos\phi) |\cos\theta|^{\varepsilon_2} |\cos\phi|^{\varepsilon_1} + r \cdot \cos\theta\cos\phi \\ Y &= b \cdot \text{sign}(\sin\theta\cos\phi) |\sin\theta|^{\varepsilon_2} |\cos\phi|^{\varepsilon_1} + r \cdot \sin\theta\cos\phi \\ Z &= c \cdot \text{sign}(\sin\phi) |\sin\phi|^{\varepsilon_1} + r \cdot \sin\phi \end{aligned} \quad (1)$$

$$-\pi/2 \leq \theta \leq \pi/2, -\pi \leq \phi \leq \pi$$

Constants  $a$ ,  $b$  and  $c$  give the extent of the ESQ. Exponents  $\varepsilon_1$  and  $\varepsilon_2$  are the shape parameters of the ESQ. The exponents are functions of  $\theta$ (azimuth) and  $\phi$ (elevation) respectively, modeled using cubic splines. An ESQ at an arbitrary position also has translation and rotation parameters. We define offset  $r$  in terms of the ESQ as  $r(\theta, \phi) = -s(\theta, \phi) \cdot ||h(\theta, \phi)||$  where, scale factor  $s(\theta, \phi)$  is modeled by RBF. This implies that the surface deformations are modeled by non-uniform scaling of the ESQ surface. It has to be noted that  $f$  is injective and its domain is closed. That is, the shapes it can model have exactly one point on the surface given an azimuth and an elevation.

The implicit form of an ESQ is given as

$$F(x, y, z) = \left[ \left( \frac{|x|}{a} \right)^{2/\varepsilon_2} + \left( \frac{|y|}{b} \right)^{2/\varepsilon_2} \right]^{\varepsilon_2/\varepsilon_1} + \left( \frac{|z|}{c} \right)^{2/\varepsilon_1} = 1. \quad (2)$$

An arbitrary point  $P$  on a shape can be expressed in terms of an ESQ as  $P = F(P)^{\frac{\varepsilon_1}{2}} \cdot Q$  where,  $Q$  is a point on the ESQ which has the same azimuth ( $\theta_P$ ) and elevation ( $\phi_P$ ) as  $P$  (see sec. 2.2). The scale factor to offset the ESQ surface so that it matches the shape at  $P$ , is given as  $s(\theta_P, \phi_P) = (1 - F(P)^{\frac{\varepsilon_1}{2}})$ . Given the implicit form  $F$  of ESQ and its relation to the

scale factor  $s$  that is modeled by RBF, the implicit form our shape model can be written as

$$F(x, y, z)^{\frac{\epsilon_1}{2}} + \sum_{j=1}^N w_j \lambda(d_j) = 1 \quad (3)$$

$\lambda$  is a Gaussian with compact support,  $d_j$  is the distance of a point  $(x, y, z)$  from the  $j^{\text{th}}$  RBF center measured as cosine distance,  $w_j$  is a weight associated with  $j^{\text{th}}$  RBF center. RBF centers are uniformly sampled on a unit sphere.

## 2.2 Shape Model Fitting

A two stage process is used for fitting the shape model to points. ESQ is first fit to the point cloud and then RBF is used to fit the residual error. The error of fit function for fitting ESQ is defined using its implicit form. The error of fit (EoF) for a point  $P$  is measured as the distance from point  $P$  to a point  $Q$  on the ESQ surface such that  $P - O = \beta(Q - O)$  where  $O$  is the center of the ESQ. Assuming that the ESQ is centered at the origin.

$$\begin{aligned} F(P/\beta) &= 1 \\ \beta &= [F(P)]^{\epsilon_1/2} \\ D &= |\beta - 1| \times \|(P/\beta)\|. \end{aligned} \quad (4)$$

The EoF is hence

$$EoF = \sum_{i=1}^{N_{data}} \|(x_i, y_i, z_i)\| \times |1 - F(x_i, y_i, z_i)^{\epsilon_1/2}| + C_1 + C_2 + C_3. \quad (5)$$

Additional constraints  $C_1$ ,  $C_2$  and  $C_3$  on the parameters of ESQ are also added to the EoF. A smoothness constraint for the exponent function is added. Since cubic splines are used as the interpolation functions for the exponents, the smoothness term penalizes the fit if the leading coefficient of the cubic polynomials are greater than a threshold  $T$ . As described in [9] additional constraints that both scale and exponent functions must be positive and that the ESQ fit is of the least possible volume are also added.

$$\begin{aligned} C_1 &= \alpha_1 [\max(f_1''' - T, 0) + \max(f_2''' - T, 0)] \\ C_2 &= \alpha_2 [\max(-a, 0) + \max(-b, 0) + \max(-c, 0) + \max(-\epsilon_1, 0) + \max(-\epsilon_2, 0)] \\ C_3 &= \alpha_3 (a * b * c) \end{aligned} \quad (6)$$

$\alpha_1$ ,  $\alpha_2$  and  $\alpha_3$  are weights for the three constraint terms respectively. Levenberg-Marquardt method for non-linear least square fit is used iteratively for minimizing the EoF and finding the number of control points required for exponent functions[9]. The weights  $\alpha_1$ ,  $\alpha_2$  and  $\alpha_3$  were set to 1,100 and 1 respectively. The threshold  $T$  was set to 500 in our experiments for fitting the shape model.

The residual error is measured as  $(1 - \beta)$  in Eq. 4. Before measuring residual error, the input point cloud is transformed using the orientation parameters of the ESQ such that it is centered at the origin and aligned to the coordinate axes. A Gaussian with compact support is used as the radial basis function. The RBF centers are picked by uniformly sampling points on a unit sphere. The RBF weights are obtained by solving the linear system  $Ax = b$ .  $A$  is a cosine distance matrix,  $b$  vector of residual errors and  $x$  is the vector of weights to be solved for. Each row of  $A$  is a vector of cosine distances from a point on the shape to the RBF centers. Each row of  $b$  is the residual error of ESQ fit for a point. We use the greedy algorithm [16] which iterates over number of centers to be used for fitting RBF.

### 3 Convex Decomposition

A method similar to [27, 33] is employed for decomposing the shape into approximately convex parts. The two step process involves first, finding pairs of points, called *mutex pairs*, which cannot belong to the same convex part. Then, finding cutting planes that divide the shape into parts that have no *mutex pairs*. Given a shape  $S = \{X \mid X \in R^3\}$ , we find the set  $M = \{(x, y) \mid x, y \in S \text{ and } x, y \text{ are mutually exclusive}\}$  and set of planes  $P$  that divide the shape into approximately convex parts.

#### 3.1 Mutex pairs

Points  $x \in S, y \in S$  are said to be mutually exclusive if there exists a plane  $p$  such that  $x, y$  lie on  $p$  and are disconnected in the contour map produced by the projection of shape  $S$  onto  $p$ . Two points  $x, y$  on the contour map are said to be connected if there exists a path  $c$  from  $x$  to  $y$  such that distance between any adjacent points along the path  $c$  is not greater than the connected-component threshold  $t$  and all the points along the path  $c$  are at the same distance from  $p$ . To find mutually exclusive points, let  $Q$  be a set of planes with normals that are evenly sampled from a unit hemisphere. Then, for each plane  $p_i$  in the set  $Q$ , a function  $f_i(S)$  is defined that projects  $S$  onto  $p_i$  and bins the distances from  $S$  to  $p_i$  to find the contour map. Connected components that are at the same distance from  $p_i$  are then estimated.  $C_{di}^j$  is the  $j^{\text{th}}$  connected component at distance  $d$  from the plane  $p_i$ ,  $n_{di}$  is the number of connected components at distance  $d$  from plane  $p_i$ . *mutex pairs* are then given by

$$M = \{(X, Y) \mid X = f_i^{-1}(x), Y = f_i^{-1}(y)\} \quad (7)$$

$$(x, y) = \underset{x \in C_{di}^u, y \in C_{di}^v}{\operatorname{argmin}} \|x - y\|$$

where,  $i = 1$  to  $|Q|$ ,  $di \in \{\text{distances from } p_i\}$ ,  $u = 1$  to  $n_{di}$ ,  $v = 1$  to  $n_{di}$  and  $u \neq v$ . There might be more than two connected components for a given distance  $d$  and plane  $p_i$ . In such cases, a *mutex pair* is deleted if the projection onto  $p_i$  of the path joining the pair passes through a connected component other than the ones the pair belong to. This can be determined by checking if there exists points on the plane that are on either side of the line segment and belong to the same connected component. A weight  $w_i$  is associated with each *mutex pair*  $M_i$ . Let  $p_1$  be the plane passing through the pair  $M_i$  that classified them as *mutex*. Consider the plane  $p_2$  which also passes through the pair  $M_i$  and is perpendicular to  $p_1$ . If the pair is connected in the contour map  $m$  produced by projecting  $S$  onto  $p_2$  then,  $w_i$  is measured as the maximum local curvature of the connected component containing the pair  $M_i$  on  $m$  else,  $w_i$  is set to zero. If repeated pairs exist in  $M$  then the one with the maximum associated weight is retained and the rest removed. It has to be noted that  $(X, Y)$  is the same as  $(Y, X)$  and that a pair of points may also be classified as mutually exclusive by multiple planes passing through them.

#### 3.2 Cutting Planes

Given set  $M$  of *mutex pairs*, we wish to find planes that divide the shape into parts that have no *mutex pairs*. For each *mutex pair* a candidate cutting plane is defined as a plane that bisects the pair and has its normal parallel to the pair. The set of candidate planes are then

given as,

$$CP = \{P_i \in R^4 \mid P_i(1:3) \cdot \left(\frac{X_i - Y_i}{\|X_i - Y_i\|}\right) = 1, P_i \cdot \left[\frac{(X_i - Y_i)}{2} \mid 1\right] = 0, (X_i, Y_i) \in M\}. \quad (8)$$

This set of planes guarantee a subdivision of the shape into parts that have no *mutex pairs*. The goal is to find the set of cutting planes  $P \subset CP$  with the minimum number of planes required to remove all *mutex pairs*. The hybrid shape model described in section 3 can accurately model approximately convex objects. Increasing the number of small parts results in representation that is no longer compact. The following heuristics are defined to find a balance between accuracy and compactness of the shape model: 1) The cut reduces the total non-convexity of the shape; 2) The cut results in parts where at least one is likely to contain no *mutex-pairs*.

The first heuristic effects the accuracy of the representation while the second effects the compactness. Formally, a plane  $P_i$  splits  $S$  into two parts  $L_1$  and  $L_2$ , let  $D$  be the set of indices of *mutex pairs* that are cut by  $P_i$ ,  $M_{L_1}$  and  $M_{L_2}$  be the *mutex pairs* that remain in  $L_1$  and  $L_2$  respectively. The utility of splitting the shape by a plane  $P_i$  is defined as

$$U(P_i) = \left(1 - \frac{\min(|M_{L_1}|, |M_{L_2}|)}{\max(|M_{L_1}|, |M_{L_2}|)}\right) \times \sum_{j \in D} \left[w_j \times \left(P_i(1:3) \cdot \frac{X_j - Y_j}{\|X_j - Y_j\|}\right)\right] \quad (9)$$

$$P_c = \underset{P_i \in CP}{\operatorname{argmax}} U(P_i).$$

The plane  $P_c$  with most utility is selected to split the shape  $S$ . This process is recursively repeated on each of the divided parts until all the *mutex pairs* are removed.

## 4 Model fitting and shape reconstruction

Given a shape as point cloud or a mesh, the method to build the hierarchical-hybrid shape representation is as follows. The input point cloud is decomposed into approximately convex parts using the method described in section 3. The goal is to have a compact but accurate shape representation. So additional conditions are added to stop the recursive decomposition of the shape. The decomposition is stopped if the average non-convexity of the *mutex pairs* in the part falls below the non-convexity threshold  $t1$ . The process is also stopped if the size (measured as the volume of the bounding box) of the part falls below the size  $t2$ . The decomposed parts are modeled using the shape model described in section 2.1. The parts that are small or could not be accurately fit with shape model, i.e., parts with fitting error greater than error-threshold  $t3$  are retained as points or meshes. During the decomposition step, each intermediate node of the tree holds the cutting plane and the 2D intersection shape that is given by the points of the shape that intersect the cutting plane. This 2D intersection shape is modeled using the 2D counterpart of the shape model described in section 2.1.

The explicit form of each shape model, leaf nodes in our tree representation, can be used to generate surface of the part it models. These parts must then be stitched together to form the entire shape. Blending of these parts is done in a bottom up fashion. To blend two parts we make use of the plane that split the two parts and the 2D intersection shape described above. We discard the region of both parts that lies within  $d$  distance from the cutting plane. This creates an empty region at the union of these parts. To fill this empty region we find corresponding points in each part that lie nearest to each other in their projections on the

**Algorithm 1** Construct HHSR

---

```

1: procedure CONSTRUCTHHSR(Shape, t1, t2, t3)
2:   if size(Shape) < t2 then
3:     return Shape
4:   [mutexs, weights] ← GetMutex(Shape)
5:   if sum(weights) > t1 then
6:     [Part1, Part2, cut] ← DecomposeShape(Shape, mutexs, weights)
7:     Node.Left ← ConstructHHSR(Part1, t1, t2, t3)
8:     Node.Right ← ConstructHHSR(Part2, t1, t2, t3)
9:     Node.Cut ← cut
10:    return Node
11:  else
12:    [ShapeModel, error] ← FitShapeModel(Shape)
13:    if error > t3 then
14:      return Shape
15:    else
16:      return ShapeModel

```

---

**Algorithm 2** Construct Shape

---

```

1: procedure CONSTRUCTSHAPE(Node)
2:   if isLeafNode(Node) then
3:     return GenerateShape(Node)
4:   Shape1 ← ConstructShape(Node.Part1)
5:   Shape2 ← ConstructShape(Node.Part2)
6:   return MergeShape(Shape1, Shape2, Node.Cut)

```

---

cutting plane. These points and their closest point on the 2D intersection shape are used as control points to fit a Bezier curve. This yields a smooth transition between the components of the shape.

## 5 Experiments and Results

To show that the representation can compactly and accurately model complex shapes, we conducted experiments on the Stanford bunny which had approximately 8000 points. For finding mutually exclusive points, the distance threshold for connected components was set to 3 mm. The non-convexity threshold to stop the decomposition of the shape was set to 5% of the total non-convexity. The shape model fitting error threshold was set to 7mm. The final representation had 937 parameters and the mean error was 1.5mm. We show the process of fitting the shape representation to the data Fig. 1.

We also conducted experiments on shapes of Prostate and Liver. The shape of the prostate was extracted from manually segmented T2-weighted 3D MR images. All the shapes were first smoothed and aligned to each other. Shapes of liver was extracted from manual segmentation of CT images [63]. To fit our shape representation, for both the shapes, we set the connected component threshold to 5 mm, non-convexity threshold to 0 and the decomposed part size threshold to 0. The shape model fitting error threshold was set to 10% of the longest

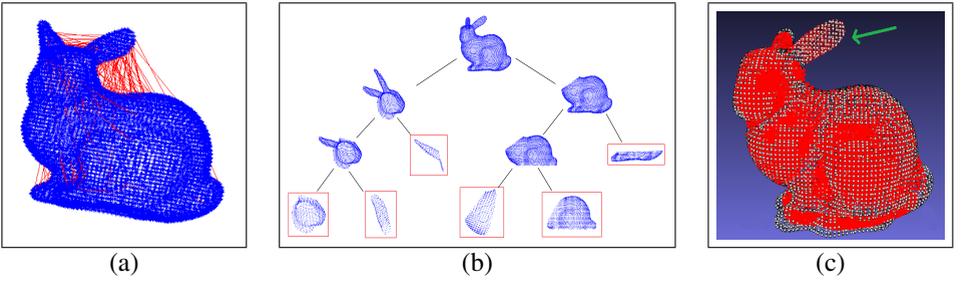


Figure 1: Fitting our shape model to Stanford Bunny (8454 vertices): (a) Mutually Exclusive pairs of points shown as red lines; (b) Approximate convex decomposition, parts shown inside red box; (c) Shown in Red is HSSR fit, ground truth boundary points are shown in White and Green arrow points to the part that was retained as points due to large error of fit of the shape model.

side of the shape.

We added 3 additional scale parameters to the shape model. This allowed each octant of the shape to have different scale parameters. An additional optimization using the LM method was added to the previously described ESQ fitting algorithm to solve for these scale parameters. We use only a subset of the input points for fitting ESQ to speed up the fitting process. The graph in Fig. 2(d) shows the effect of number of points used on the accuracy of ESQ fit. We see that the number of points used to fit ESQ to the data has negligible effect on the mean error. Fig. 2(a), 2(b), 2(c) show the effect of number of points used for fitting on the ESQ parameter values, which is negligible. We could achieve roughly the same parameter values from 925 points and from 125 uniformly sampled points.

We compared the compactness and accuracy of our representation with spherical harmonics representation of the shape,[52] and RBF iso-surface which is fit using method described in[52]. To fit RBF iso-surface we used a greedy approach and iterated over the number of control points and stopped when the error was comparable to the error of HSSR. Table 1 shows the results.

	Method	Mean Error (in mm)	Number of Parameters
Prostate	HSSR(ours)	$0.152 \pm 0.067$	303-337
	RBF iso-surface [52]	$0.222 \pm 0.041$	1536
	Spherical Harmonics [52]	$0.539 \pm 0.405$	1089
	Method	Mean Error (in mm)	Number of Parameters
Liver	HSSR(ours)	$1.116 \pm 0.212$	788-989
	RBF iso-surface [52]	$3.028 \pm 0.296$	4952
	Spherical Harmonics [52]	$8.265 \pm 2.22$	4225

Table 1: Mean fitting error and number of parameters used for 45 prostate shapes and 10 liver shapes using different methods of shape representation. HSSR has more accurate fit with fewer parameters

From Table. 1, our HSSR shape model has much smaller error than the other two methods, while using much smaller number of parameters ( $\approx 20\%$  of RBF iso-surface and less

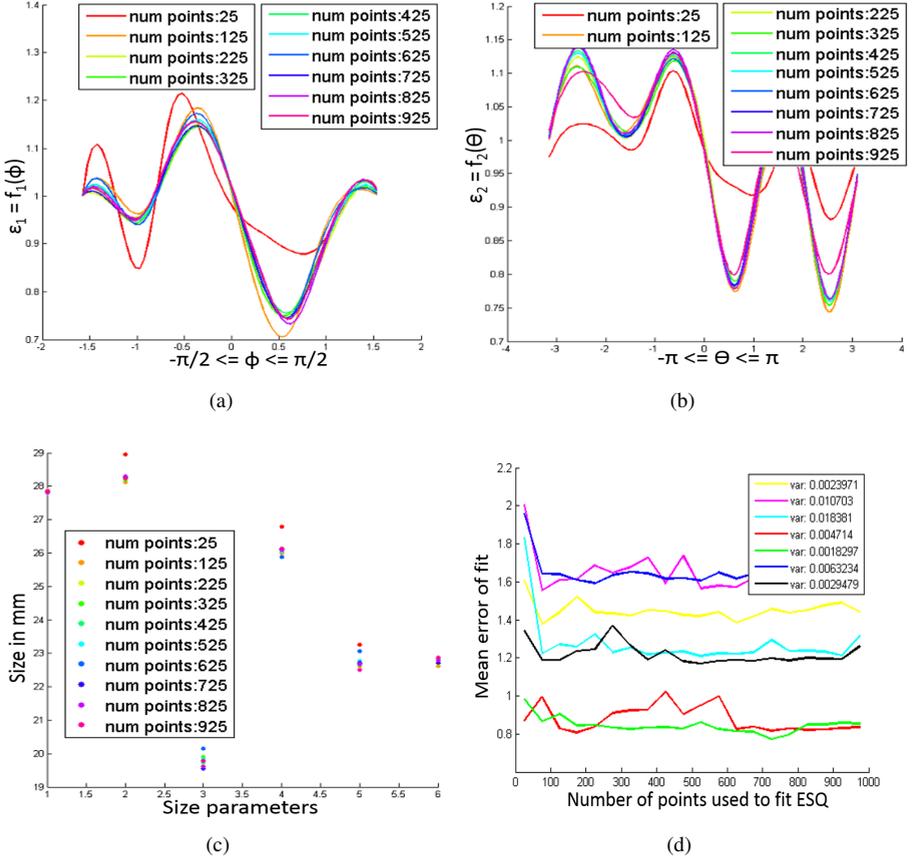


Figure 2: Effect of the number of points used to fit ESQ. (a) Effect on the shape parameter  $\varepsilon_1$  which is a function of elevation; (b) Effect on the shape parameter  $\varepsilon_2$  which is a function of azimuth; (c) Effect on the 6 size parameters; (d) Effect of the number of points used on accuracy of ESQ fit. Each curve represents a sample shape

than 30% of Spherical Harmonics on prostate). Our model enjoys an increasing benefit when the shapes get more complex, as indicated in the liver data. The variation in the number of parameters required to fit the shapes indicates the variations in shapes of the organs. We can notice from Fig. 3 that our shape models fit the ground truth of Prostate and Liver well.

The shape model used has both implicit and explicit forms. The implicit form can also function as an inside/outside function and can be used to test if a point is within or outside the shape. Using HSSR, this test can be performed by first finding the section of the shape that the point lies. This is done by testing it with the intersection planes at each level of the tree. Once the position of the point in the tree is determined, the implicit form of the shape model can be used to test where the point is with respect to the shape. In cases where the point lies close to the intersection plane, the implicit form of intersection shape is used. This is especially useful for image guided interventions. Given the trajectory of a surgical device, this inside/outside test can be used to determine the region of the organ the device will come

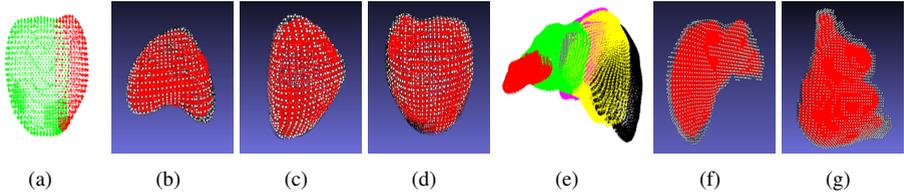


Figure 3: Shape representation for Prostate and Liver shapes. (a) Prostate split into approximately convex parts; (b), (c) and (d) are axial, sagittal and coronal views of prostate. (e) Points generated from shape model that was fit to each split part of a liver shape. (f) and (g) are different view of the same liver shape. Red represents the mesh generated form HSSR. Shown in White are the ground truth organ boundary points.

in contact with, using the bisection method recursively for the points that lie on the trajectory of the device. In each recursion the part of the trajectory that is completely inside the shape can be removed (Fig. 4).

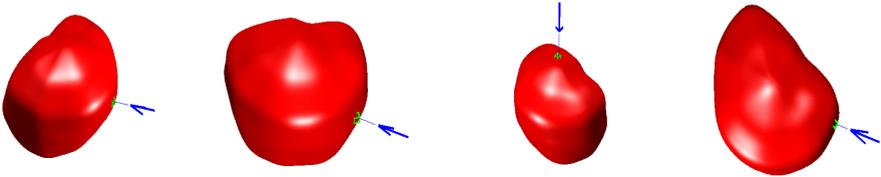


Figure 4: Intersection of an arbitrary line with the surface of the shape, the point of intersection is shown in green.

## 6 Conclusion

In our work we present a hybrid shape model which has both implicit and parametric forms and can model a wide range of shapes. An automatic method to fit this shape model to complex medical shapes by hierarchically dividing the shape into approximately convex parts is also presented. Finally, we also present a method to blend the hybrid representation of the parts to reconstruct the entire shape. We show that the HSSR is able to compactly and accurately model complex medical shapes. We show that the representation can be used to guide surgical procedures by providing the ability to track surgical device trajectory with respect to the shape.

## Acknowledgment

We would like to thank Dr. Ismail Baris Turkbey of the Molecular Imaging Program at the Center for Cancer Research, National Institute of Health, MD, USA for providing the data and domain expertise.

## References

- [1] Krieger, Axel, Iulian I. Iordachita, Peter Guion, Anurag K. Singh, Aradhana Kaushal, Cynthia Menard, Peter A. Pinto, Kevin Camphausen, Gabor Fichtinger, and Louis L. Whitcomb, "An MRI-Compatible Robotic System With Hybrid Tracking for MRI-Guided Prostate Intervention", *IEEE Transactions Biomedical Engineering*, 2011.
- [2] Robert Susil, Axel Krieger, and Andrew Derbyshire, Attila Tanacs, Louis Whitcomb, Gabor Fichtinger, Ergin Atalar, "System for MR Image guided Prostate Interventions: Canine Study", *Radiology*, 2003.
- [3] Patriciu A, Petrisor D, Muntener M, Mazilu D, Schaer M, Stoianovici D. Automatic brachytherapy, "Seed placement under MRI guidance", *IEEE Transactions on Biomedical Engineering*", 2007.
- [4] Stoianovici, Dan, Danny Song, Doru Petrisor, Daniel Ursu, Dumitru Mazilu, Michael Mutener, Michael Schar, and Alexandru Patriciu, "MRI Stealth robot for prostate interventions", *Minimally Invasive Therapy and Allied Technologies*, 2007.
- [5] Muntener, Michael, Alexandru Patriciu, Doru Petrisor, Michael Schar, Daniel Ursu, Danny Y. Song, and Dan Stoianovici, "Transperineal prostate intervention: robot for fully automated MR imaging system description and proof of principle in a canine model", *Radiology*. 2008.
- [6] Lin Zhou, and Chandra. Kambhamettu. "Extending superquadrics with exponent functions: modeling and reconstruction", *IEEE Conference on Computer Vision and Pattern Recognition*, 1999.
- [7] Talib S Bhabhrawala and Venkat N. Krovi and Frank C. "Mendel and Venugopal Govindaraju Shape Recovery Using Extended Superquadrics", *Technical report*, 2007.
- [8] Franc Solina, Ruzena Bajcsy, Arqđ Ruzena Bajcsy, "Recovery of Parametric Models from Range Images: The Case for Superquadrics with Global Deformations", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1990.
- [9] Ismail B. re, Sayan D. Pathak, Yongmin Kim, "3D prostate shape modeling from sparsely acquired 2D images using deformable models", *SPIE Conference on Visualization, Image-Guided Procedures, and Display on Medical Imaging*, 2004.
- [10] Ismail B. Tutar, Sayan D. Pathak, Lixin Gong, Paul S. Cho, Kent Wallner, and Yongmin Kim, "Semiautomatic 3-D Prostate Segmentation from TRUS Images Using Spherical Harmonics", *IEEE transactions on Medical Imaging*, 2006.
- [11] Eric Bardinet, N. Ayache, and L. D. Cohen, "Fitting of Iso-Surfaces Using Superquadrics and Free-Form Deformations", *IEEE Workshop on Biomedical Image Analysis*, 1994.
- [12] Knoll, Aaron, Younis Hijazi, Andrew Kensler, Mathias Schott, Charles Hansen, and Hans Hagen "Fast Ray Tracing of Arbitrary Implicit Surfaces with Interval and Affine Arithmetic", *Computer Graphics forum*, 2009.
- [13] Devendra Kalra, Alan Barr, "Guaranteed Ray Intersections with Implicit Surfaces", *Computer Graphics*, 1989.

- [14] Joshua B. Tenenbaum, Vin de Silva, John C. Langford, "A Global Geometric Framework for Nonlinear Dimensionality Reduction", *Science*, 2000.
- [15] Bryan Morse, Terry S. Yoo, Penny Rheingans, David Chen, and Kalpathi Subramanian, "Interpolating implicit surfaces from scattered surface data using compactly supported radial basis functions", *ACM SIGGRAPH*, 2005.
- [16] Carr, Jonathan C., Richard K. Beatson, Jon B. Cherrie, Tim J. Mitchell, W. Richard Fright, Bruce C. McCallum, and Tim R. Evans, "Reconstruction and representation of 3D objects with radial basis functions", 28th annual ACM conference on Computer graphics and interactive techniques, 2001.
- [17] Cootes Timothy, Christopher Taylor, "Combining point distribution models with shape models based on finite element analysis." *Image and Vision Computing*, 1995.
- [18] Terzopoulos, Demetri, and Dimitri Metaxas. "Dynamic 3D models with local and global deformations: Deformable superquadrics." *International Conference on Computer Vision*, 1990
- [19] Tobias Heimann, and Hans-Peter Meinzer. "Statistical shape models for 3D medical image segmentation: A review." *Medical image analysis*, 2009.
- [20] Lixin Gong, Pathak, S.D., Haynor, D.R., Cho, P.S., Yongmin Kim, "Parametric shape modeling using deformable superellipses for prostate segmentation," *IEEE Transactions on Medical Imaging*, 2004.
- [21] Jayadevappa, D., S. Srinivas Kumar, and D. S. Murty. "Medical Image Segmentation Algorithms using Deformable Models: A Review." *IETE Technical Review*, 2011.
- [22] Langs Georg, Allan Hanbury, Bjoern Menze, and Henning Muller. "VISCERAL: Towards large data in medical imaging challenges and directions", *Medical Content-Based Retrieval for Clinical Decision Support*. 2013.
- [23] Erdman Arthur, Daniel Keefe, Randall Schiestl, "Grand challenge: Applying regulatory science and big data to improve medical device innovation." *IEEE Transactions on Biomedical Engineering*, 2013.
- [24] Stephan Bischoff, Leif Kobbelt, "Ellipsoid decomposition of 3D-models," *International Symposium on 3D Data Processing Visualization and Transmission*, 2002.
- [25] Aaron Lee, Henry Moreton and Hugues Hoppe, "Displaced Subdivision Surfaces", *Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2000*.
- [26] Hairong Liu, Wenyu Liu, Latecki, L.J., "Convex shape decomposition," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.
- [27] Henrik Zimmer, Campen Marcel, Leif Kobbelt. "Efficient Computation of Shortest Path-Concavity for 3D Meshes." *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013.
- [28] Andrew Hanson, "Hyperquadrics: smoothly deformable shapes with convex polyhedral bounds." *Computer vision, graphics, and image processing*, 1988.

- [29] Laurent Chevalier, Fabrice Jaillet, Atilla Baskurt, "Segmentation and Superquadric Modeling of 3D Objects", Journal of WSCG, 2003.
- [30] Isaac Cohen, Laurent D. Cohen, "A Hybrid Hyperquadric Model for 2-D and 3-D Data Fitting", Computer Vision and Image Understanding, 1996.
- [31] Zhou Ren, Junsong Yuan, Chunyuan Li, Wenyu Liu, "Minimum near-convex decomposition for robust shape representation," IEEE International Conference on Computer Vision (ICCV), 2011.
- [32] Van Nguyen, Hien, and Fatih Porikli. "Support vector shape: A classifier-based shape representation." IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013.
- [33] T. Heimann, B. van Ginneken, M. Styner, Y. Arzhaeva, V. Aurich, C. Bauer, A. Beck, C. Becker, R. Beichel, G. Bekes, F. Bello, G. Binnig, H. Bischof, A. Bornik, P.M.M. Cashman, Y. Chi, A. Cordova, B.M. Dawant, M. Fidrich, J. Furst, D. Furukawa, L. Grenacher, J. Hornegger, D. Kainmueller, R.I. Kitney, H. Kobatake, H. Lamecker, Th. Lange, J. Lee, B. Lennon, R. Li, S. Li, H.-P. Meinzer, G. Nemeth, D.S. Raicu, A.-M. Rau, E.M. van Rikxoort, M. Rousson, L. Rusko, K.A. Saddi, G. Schmidt, D. Seghers, A. Shimizu, P. Slagmolen, E. Sorantin, G. Soza, R. Susomboon, J.M. Waite, A. Wimmer, I. Wolf, "Comparison and Evaluation of Methods for Liver Segmentation from CT datasets", IEEE Transactions on Medical Imaging, volume 28, number 8, pp. 1251-1265, 2009.
- [34] Wu Jia, Marc Simon, and John C. Brigham. "A comparative analysis of global shape analysis methods for the assessment of the human right ventricle." Computer Methods in Biomechanics and Biomedical Engineering: Imaging and Visualization, 2014.