Computer Science Department of The University of Auckland
CITR at Tamaki Campus (http://www.citr.auckland.ac.nz)

CITR-TR-104 November 2001

An Approximative Calculation of Relative Convex Hulls for Surface Area Estimation

Linjiang Yu and Reinhard Klette[1]

Abstract

Relative convex hulls have been suggested for multigrid-convergent surface area estimation. Besides the existence of a convergence theorem there is no efficient algorithmic solution so far for calculating relative convex hulls. This article discusses an approximative solution based on minimum-length polygon calculations. It is illustrated that this approximative calculation also proves (experimentally) to provide a multigrid convergent measurement.

[1] Center for Image Technology and Robotics Tamaki Campus, The University of Auckland, Auckland, New Zealand. lyu011@ec.auckland.ac.nz and r.klette@auckland.ac.nz

An Approximative Calculation of Relative Convex Hulls for Surface Area Estimation

Linjiang Yu and Reinhard Klette*

Abstract

Relative convex hulls have been suggested for multigrid-convergent surface area estimation. Besides the existence of a convergence theorem there is no efficient algorithmic solution so far for calculating relative convex hulls. This article discusses an approximative solution based on minimum-length polygon calculations. It is illustrated that this approximative calculation also proves (experimentally) to provide a multigrid convergent measurement.

Keywords: *digital geometry, surface measurement, global polyhedrization, relative convex hull*

1. Introduction

Surface area is one of the important three-dimensional (3D) shape features in computer-based image analysis, and its definition and calculation has been discussed in mathematics over more than one hundred years. Recently many papers on theoretical and algorithmic aspects of surface area estimation have been published for 3D digital objects. Polyhedrization approaches are often used for approximation of surface area, and these are also efficient ways to visualize a surface.

The multigrid convergence problem for surface area estimation for 3D digital objects may be stated as follows [1]: assume a measurable solid in 3D Euclidean space being digitized with respect to finer and finer grid resolution. The resulting digital object is used as input for a surface area estimation program, calculated estimates should converge to a fixed value assuming finer and finer grid resolution, and this fixed value should be the true surface area. In the context of image analysis this may be stated as follows:

1. Image acquisition at higher grid resolutions should lead to convergence for the value of surface area.
2. Convergence should be towards the true value.

*CITR Tamaki, University of Auckland Tamaki Campus, Building 731, Auckland, New Zealand {*lyu011@ec.auckland.ac.nz, r.klette@cs.auckland.ac.nz*}

The paper [2] proposed a way to classify polyhedrization techniques and algorithms. It expresses a general hypothesis that local polyhedrization techniques such as surface tracking (opaque cubes), marching cubes, marching tetrahedra etc. are failing to meet both convergence properties, and global polyhedrization techniques such as convex hull computation (for digitization of convex sets), digital planar segmentations, etc. are likely to meet both convergence properties.

2. Relative Convex Hull (RCH) Theorem

The notion of a relative convex hull (RCH) is proposed in [3]. Let $V \subseteq U \subseteq R^3$. We define a set $conv_U(V)$ which will be the relative convex hull of V with respect to set U.

Definition 1 *Let $U \subseteq R^3$ be an arbitrary set. A set $C \subseteq U$ is said to be* U-convex *iff for every $x, y \in C$ such that $\overline{xy} \subseteq U$ it holds that $\overline{xy} \subseteq C$.*

Definition 2 *Let $V \subseteq U \subseteq R^3$ be given. The intersection of all U-convex sets containing V is called the* U-convex hull *of V and denoted by $conv_U(V)$.*

Let $conv(V)$ be the convex hull of set V. Three straightforward conclusions:

1. $V \subseteq conv_U(V) \subseteq U \cap conv(V)$.
2. $V = conv_U(V)$ iff V is *U-convex*.
3. $conv_U(V) = conv(V)$ iff $conv(V) \subseteq U$.

We consider a polyhedral approximation of a surface $\partial\Theta$ provided that the set $\Theta \subset R^3$ has a measurable surface area $s(\Theta)$ defined in the Minkowski sense [5]. A bounded set Θ is a Jordan set iff $S = \partial\Theta$ is homeomorphic to the unit sphere. Let $U = J_r^+(\Theta)$, $V = J_r^-(\Theta)$ be the outer and inner digital sets of set Θ, see, e.g., [1, 3]. These digitizations are defined with respect to a regular orthogonal grid with grid resolution $r > 0$. Then [3]

$$CH_{J_r^+(\Theta)}(J_r^-(\Theta)) = \partial conv_{J_r^+(\Theta)}(J_r^-(\Theta))$$

is a closed polyhedral surface, and we call it *relative convex hull* (RCH in brief). The multigrid convergence of surface estimations based on relative convex hulls is ensured by the following theorem [3]:

Theorem 1 *Let $\Theta \subseteq R^3$ be a smooth Jordan set. Let $J_r^+(\Theta)$, $J_r^-(\Theta)$, $r \geq 0$, be digitizations of this set. Then*

$$\lim_{r\to\infty} s(CH_{J_r^+(\Theta)}(J_r^-(\Theta))) = s(S),$$

where s(.) is defined in the Minkowski sense.

Following this theorem, surface area estimation of 3D digital objects can meet both multigrid convergence constraints using the RCH polyhedrization approach. However, the algorithmic treatment of relative convex hulls remains to be an open problem yet.

For digital objects in 2D space, the minimum length polygon (MLP) is uniquely defined, and the length estimation of digital curves can meet both multigrid convergence constraints using the MLP method (see, e.g. the paper [4] for a linear-time MLP algorithm).

This paper suggests an approximative solution for RCH calculations based on MLP calculations by slicing surfaces of digital objects into 2D digital curves, calculating MLPs in 2D for each of the digital curves, then accumulating the surface area by connecting vertices of MLPs of the digital curves in a certain manner. The class of input sets needs to be restricted for this approximative estimation procedure.

3. Principle of MLP Approximation

Consider a bounded, closed subset Θ of Euclidean plane in 2D space, its inner digital set $J_r^-(\Theta)$ and outer digital set $J_r^+(\Theta)$ with respect to a grid resolution r. The MLP [4] lies completely in the open r-boundary $J_r^+(\Theta)\setminus(J_r^-(\Theta) \cup \partial J_r^+(\Theta))$ of set Θ and circumscribing $\partial J_r^-(\Theta)$.

Assume X-Z coordinate system, $+Z$ axis points upwards and $+X$ axis points to the right. The sign of turn of a vertex P_i is decided by the determinant value det(P_{i-1}, P_i, P_{i+1}), where P_i=(x_i, z_i). Trace a frontier $\partial J_r^-(\Theta)$ in counterclockwise direction, put the non-zero turn vertices into a list in order. Replace each negative vertex in this list by its corresponding negative vertex of $\partial J_r^+(\Theta)$ simply by modifying the coordinates by ± 1 according to the direction of the incident edges. The resulting list contains all the vertices which may become vertices of the MLP. A vertex of the list may become an MLP-vertex if the following conditions are fulfilled. Suppose that the ith vertex P_i of the list is an MLP-vertex. Another vertex P_j, with $j > i$, may belong to the MLP if (see Figure 1)

- All positive vertices P_k^+ with $i < k < j$ lie on the positive side of (P_i, P_j) or are collinear with it, i.e. det(P_i, P_k^+, P_j)≥0 holds.

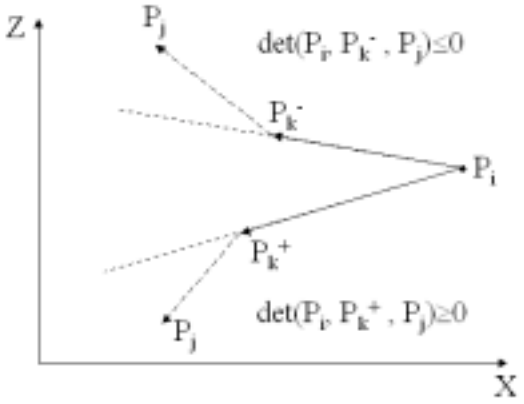

Figure 1. Suppose that the ith vertex P_i of the list is an MLP-vertex, another vertex P_j, with $j > i$, may belong to the MLP if det(P_i, P_k^+, P_j)≥0 or det(P_i, P_k^-, P_j)≤0 with $i < k < j$.

- Or all negative vertices P_k^- with $i < k < j$ lie on the negative side of (P_i, P_j) or are collinear with it, i.e. det(P_i, P_k^-, P_j)≤0 holds.

Suppose both a positive vertex P^+ and a negative vertex P^- satisfy the above conditions. Three situations may occur by testing next vertex P (see Figure 2):

- Vertex P^+ becomes the next MLP vertex if det(P_i, P^+, P)>0.
- Vertex P^- becomes the next MLP vertex if det(P_i, P^-, P)<0.
- Vertex P becomes a candidate for MLP and must replace either P^+ or P^- corresponding to the sign of P, otherwise.

Figure 3 illustrates an example of MLP calculation for a digital set $J_r^-(\Theta)$ in 2D space using linear-time MLP algorithm in the paper [4]. Black dots represent negative vertices, white dots represent positive vertices, the polygon connected with black lines is the resulting MLP.

4. Relative Convex Hull (RCH) Approximation

Assume an orthogonal Cartesian coordinate system X-Y-Z. Let $\Theta \subseteq \mathcal{R}^3$ be a bounded set. The digital space $\mathcal{Z}^3$ is represented using r-grid cubes $C^r_{i,j,k}$, and $\{i, j, k\}$ is the centroid of $C^r_{i,j,k}$, with six r-faces parallel to the coordinate planes, with r-edges of length $1/r$, for resolution parameter $r \geq 1$ and integers i, j, k with $0 \leq i < dim_x$, $0 \leq j < dim_y$ and $0 \leq k < dim_z$. We consider a digital

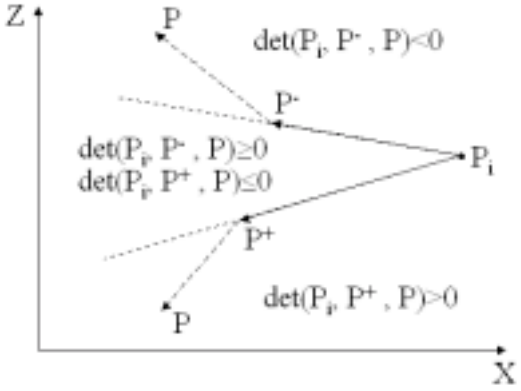

Figure 2. Suppose both a positive vertex P^+ and a negative vertex P^- satisfy the conditions in Figure 1. Three situations may occur by testing next vertex P.

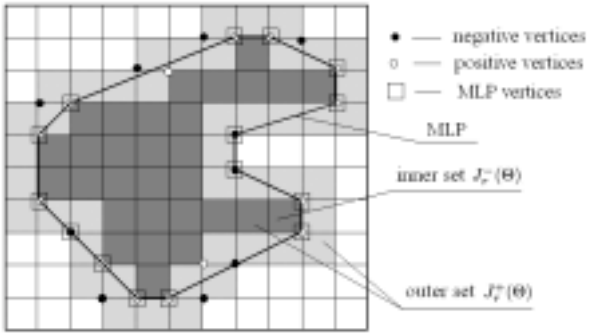

Figure 3. Example of MLP calculation for a digital set $J_r^-(\Theta)$ in 2D space using linear-time MLP algorithm in the paper [4].

object $J_r^-(\Theta)$ in $\mathcal{Z}^3$ defined by the Jordan inner digitization, consisting of all r-cubes completely contained in the interior of Θ. The surface area of Θ will be approximated based on $J_r^-(\Theta)$ (i.e. a given digital object is assumed to be a set of this type). A *border r-face* along position (i,j) of $J_r^-(\Theta)$ in $+Z$ axis direction is defined in the following definition, and similar border faces are defined for $-Z$, $+Y$, $-Y$, $+X$, or $-X$ axis directions.

Definition 3 *A* border r-face $F_{i,j}$ *along position (i,j) in $+Z$ axis direction is a shared r-face between two r-cubes $C^r_{i,j,k}$ and $C^r_{i,j,k+1}$ where $C^r_{i,j,k}$ is in $J_r^-(\Theta)$, and $k+1 = dim_z$ or $C^r_{i,j,k+1}$ is not in $J_r^-(\Theta)$. The position (i,j) represents a pair of values on X and Y coordinates with $0 \le i < dim_x$, $0 \le j < dim_y$.*

According to this definition, it is possible to have zero or more border r-faces along position (i,j) in $+Z$ axis direction for a surface of a digital set $J_r^-(\Theta)$, e.g. an object that possesses cavities may have more than one border r-face along a certain position.

The surface of a digital set $J_r^-(\Theta)$ can be partioned into *six disjoint axial manifolds* in their corresponding axial directions, which might be indexed with $+Z$, $-Z$, $+Y$, $-Y$, $+X$, and $-X$. Let us define an *orthogonally completely visible surface*.

Definition 4 *A surface of a digital set $J_r^-(\Theta)$ is called an* orthogonally completely visible surface *iff there exists at most one border r-face along any position, for each of the six axial manifolds.*

Each border r-face $F_{i,j}$ is defined by four corner points P_0, P_1, P_2, P_3 if $F_{i,j}$ exists; otherwise let $F_{i,j} = \emptyset$. See Figure 4.

Our algorithm is only applicable to an orthogonally completely visible surface. All digitally convex objects and some non-convex objects possess orthogonally completely visible surface. Assume that a digital set $J_r^-(\Theta)$ possesses an orthogonally completely visible surface, the surface area estimation is the sum of all areas of all these six axial manifolds. For the sake of describing our approximative RCH algorithm simply and clearly, we will explain the area estimation for the $+Z$-axial manifold, and processes are similar for the other axial manifolds. The 8-neighborhood of a border r-face $F_{i,j}$ is defined as follows:

Definition 5 *The* 8-neighborhood *of a border r-face $F_{i,j}$ is the set $N_8((i,j))$ of border r-faces $F_{x,y}$ such that $max\{|x-i|, |y-j|\} = 1$.*

Each corner point P_c (0≤c≤3) of a border r-face $F_{i,j}$ may have a maximum of three corresponding corner points on $N_8((i,j))$ such that they have the same values on X and Y axes. We define these corner points as *corresponding corner points* of P_c on $F_{i,j}$. Figure 4 illustrates a corner point P_3 of a border r-face $F_{i,j}$ and its three corresponding corner points on $N_8((i,j))$, i.e. P_0 of $F_{i,j-1}$, P_1 of $F_{i+1,j-1}$, and P_2 of $F_{i+1,j}$.

The digital space $\mathcal{Z}^3$ is divided into dim_y slices along the Y-axis. Each slice Π_y, for $0 \le y < dim_y$, consists of all grid cubes $C^r_{i,y,k}$, with $0 \le i < dim_x$ and $0 \le k < dim_z$. Two border r-faces are edge-connected if they are 8-neighbors.

Definition 6 *A* face run *is an edge-connected component of border r-faces within a slice of an axial manifold in $\mathcal{Z}^3$.*

A slice Π_y may contain zero or more face runs, and a single face run may consist of just one r-face. Each face

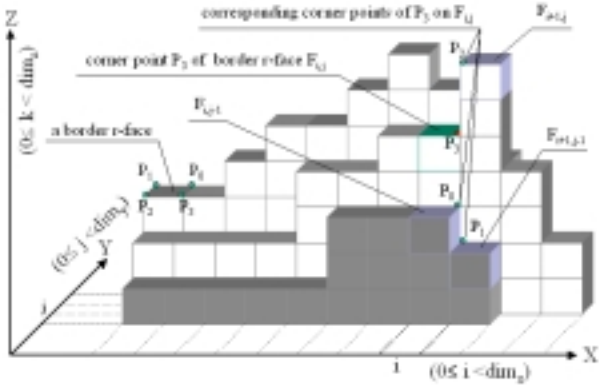

Figure 4. The corner point P_3 of a border r-face $F_{i,j}$ and its three corresponding corner points on $N_8((i,j))$, i.e. P_0 of $F_{i,j-1}$, P_1 of $F_{i+1,j-1}$, and P_2 of $F_{i+1,j}$.

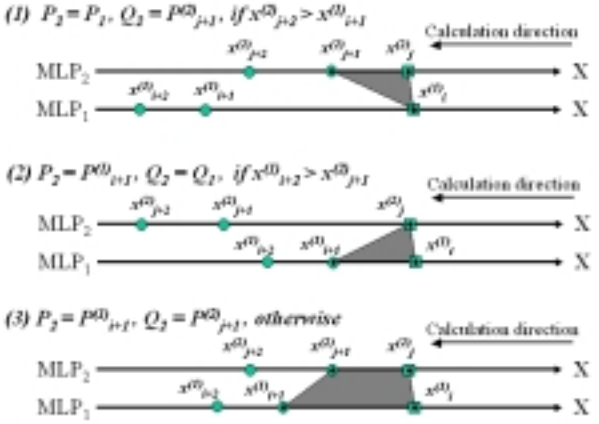

Figure 5. Three situations of triangulation of a MLP segment for a +Z axial manifold.

run in the $+Z$ axial manifold defines two 4-connected vertex sequences (one on each side of the face run) in the X-Z plane specifying sequences of potential vertices of an MLP (see the paper [4]) approximating this face run. Each vertex of the 4-connected vertex sequences is a corner point of a border r-face by replacing the value on Z axis by the maximum value on Z axis among its corresponding corner points and itself.

Our approximative relative convex hull (RCH) algorithm consists of the following steps for a $+Z$ axial manifold:

1. The digital space $\mathcal{Z}^3$ is divided into dim_y slices along the Y-axis.
2. For each slice in $\mathcal{Z}^3$, if there exists any face run, go to the next step. Otherwise, do nothing for this slice.
3. For each face run, obtain both 4-connected vertex sequences in the X-Z plane which trace through potential vertices of the MLP.
4. Use the 2D MLP algorithm to calculate an MLP segment for these two sequences. Then triangulate vertices of the MLP segment and accumulate all triangle areas into a resulting surface area value for the $+Z$ axial manifold.
5. If all slices are finished, return the surface area value for the $+Z$ axial manifold.

Now let us discuss the triangulation of a MLP segment for a +Z axial manifold. Let ($P_0^{(1)}$, $P_1^{(1)}$, $\cdots$, $P_i^{(1)}$, $\cdots$, $P_m^{(1)}$) and ($P_0^{(2)}$, $P_1^{(2)}$, $\cdots$, $P_j^{(2)}$, $\cdots$, $P_n^{(2)}$) be the vertex sequences of two MLPs, i.e. MLP_1 and MLP_2. Two pairs of points (P_1, Q_1) and (P_2, Q_2) taken from vertices of MLP_1 and MLP_2 become a calculating unit of area. Let $P_k^{(1)}$=($x_k^{(1)}$, $y_k^{(1)}$, $z_k^{(1)}$) represent a vertex of a MLP where $x_k^{(1)}$, $y_k^{(1)}$, $z_k^{(1)}$ are the values on X, Y, Z coordinates of the vertex $P_k^{(1)}$ respectively, and $P_k^{(2)}$=($x_k^{(2)}$, $y_k^{(2)}$, $z_k^{(2)}$) represent a vertex of a MLP where $x_k^{(2)}$, $y_k^{(2)}$, $z_k^{(2)}$ are the values on X, Y, Z coordinates of the vertex $P_k^{(2)}$ respectively. The process of triangulation will be done in order, one after another, on the two vertex sequences. Let P_1=$P_i^{(1)}$, Q_1=$P_j^{(2)}$; the next pair P_2, Q_2 depends on the following situations (see Figure 5):

1. P_2 = P_1, Q_2 = $P_{j+1}^{(2)}$, if $x_{j+2}^{(2)} > x_{i+1}^{(1)}$.
2. P_2 = $P_{i+1}^{(1)}$, Q_2 = Q_1, if $x_{i+2}^{(1)} > x_{j+1}^{(2)}$.
3. P_2 = $P_{i+1}^{(1)}$, Q_2 = $P_{j+1}^{(2)}$, otherwise .

An example of a triangulation of a MLP segment for a +Z axial manifold is demonstrated in Figure 6. Consider a face run of the middle slice Π_y, square marks and dot marks represent the vertex sequences of two MLPs, i.e. MLP_1 and MLP_2 respectively. The resulting triangulation consists of the black lines between vertices of the two MLPs, and the sum of areas of all the triangles is the area of the face run.

Note that a face run can be a very long sequence of border r-faces if the grid resolution r goes to infinity. Our algorithm uses a sliding window on each slice limiting the length of face runs according to memory limitations in the program. This allows to run this algorithm on digital sets of any resolution r. The window size however can be any integer specifying our algorithm as being a global polyhedrization algorithm. The computational complexity in this algorithm

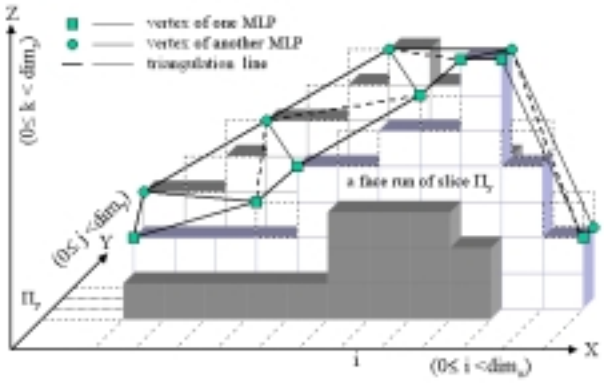

Figure 6. Example of triangulation of a MLP segment for a +Z axial manifold.

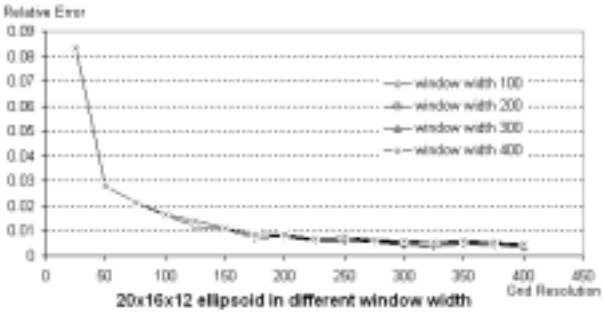

Figure 7. Impact of different widths of the sliding window on relative errors of estimated surface areas versus grid resolution for an ellipsoid with semi-axes 20×16×12 (orientation parallel to coordinate axes).

is $\mathbf{O}(n^3)$ where n is the maximum diameter of the digital object under grid resolution r. No attempts have been made at this point for optimizing the time complexity of the algorithm.

5. Experimental Results

General ellipsoids with semi-axes a, b, c, cuboids and a non-convex object are used as test objects. The relative error $\mathbf{E}_{rel}$ is used to analyze and evaluate the convergence of our polyhedrization algorithm. Let S_e be the surface area estimation of Θ and S_t be the true value of surface area of $\partial\Theta$, then the relative error $\mathbf{E}_{rel}$ is defined as

$$\mathbf{E}_{rel} = \frac{|S_e - S_t|}{S_t}.$$

Figure 7 illustrates the surface area estimation of an ellipsoid with semi-axes 20×16×12 (axes parallel to coordinate axes) if different widths w are used for the sliding window in our RCH algorithm. The widths $w = 100, 200, 300, 400$ have been used in this experiment, and results show that all curves illustrate similar multigrid convergence behaviour. The maximum average of the absolute deviations from the mean is 0.0009. Variations in width w of the moving window do not make a big difference for this algorithm. A reason may be that the resolution (or the size of the object) is not yet large enough with respect to the window size.

Figure 8 illustrates the impact of different orientations of the same ellipsoid on curves of relative errors of estimated surface areas versus grid resolution: rotating $45°$ about Z-axis, rotating $45°$ about Z- then Y-axis, and no rotation. The maximum average of the absolute deviations from the mean is 0.0037. It obviously shows that all error curves match both multigrid convergence constraints, and variations of object orientations do not make a difference for the estimated surface area using our algorithm.

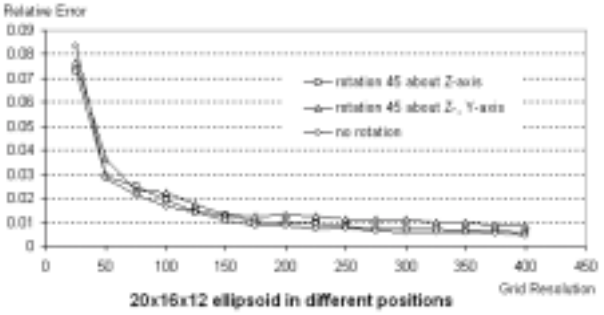

Figure 8. Relative errors of surface area estimation versus grid resolution in three different positions.

Figure 9 illustrates the impact of different object shapes on the speed of convergence. These curves show relative errors of estimated surface areas within a family of different objects, for four different grid resolutions. The family of objects are ellipsoids with semi-axes 20×20×t in orientation parallel to the coordinate axes, where parameter t is the thickness of the ellipsoid ranged from 2 to 20, i.e. from a 'flat round plate' to a sphere. From this figure we can conclude that all curves have the same trend to converge towards the true value of the surface area with increasing of resolution, but the convergence speed for flat shapes is faster than for a sphere.

Figure 10 illustrates a curve of relative errors of esti-

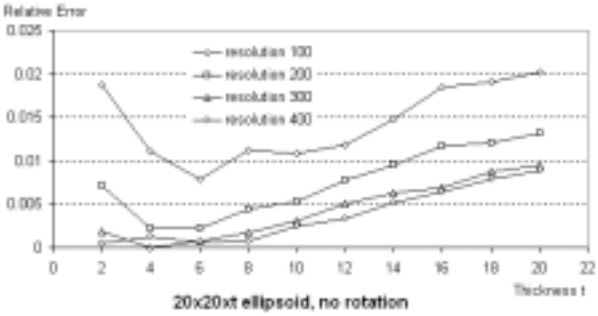

Figure 9. Relative errors of surface area estimation versus a family of objects in different grid resolution.

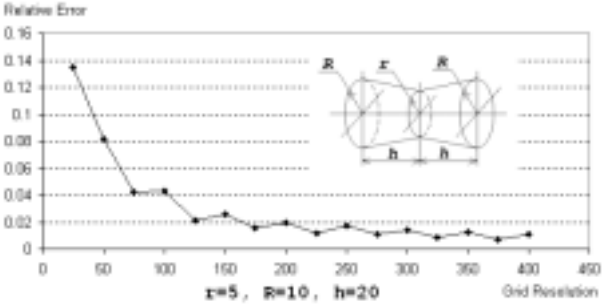

Figure 11. Relative errors of surface area estimation versus grid resolution for a non-convex object.

mated surface area versus grid resolution for a cuboid object in three different orientations: rotating $45°$ about Z-axis, rotating $45°$ about Z- then Y-axis, and no rotation. The test object is a cuboid with semi-axes $5\times4\times3$. These curves obviously show a trend to converge towards the true value of surface area.

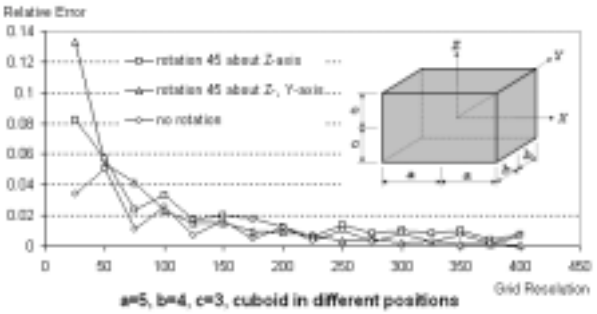

Figure 10. Relative errors of surface area estimation versus grid resolution for a cuboid object in three different orientations.

Finally, Figure 11 illustrates a curve of relative errors of estimated surface area versus grid resolution for a non-convex object. The test object is a composition of two blocks of frustum of a right circular cone connecting together with their small circular bases. The size of the frustum of the right circular cone is: r=5, R=10, h=20. This curve shows a trend to converge towards the true value of surface area, however with a little bit oscillation.

6. Conclusions

From above experimental results we conclude that our approximative calculation of relative convex hulls is a possible approach for surface area estimation. However, the object needs to have an orthogonally completely visible surface. The described approximative RCH algorithm has time complexity $\mathbf{O}(n^3)$ where n is the maximum diameter with respect to a chosen grid resolution r.

7. Acknowledgement

Thanks to Garry Tee (Auckland University) for providing a program to calculate the surface area of an arbitrary ellipsoid.

References

[1] R. Klette, Approximation and representation of 3D objects, *Advances in Digital and Computational Geometry*, Springer, Singapore (1998) 161–194.

[2] Y. Kenmochi and R. Klette, Surface area calculation for digitized regular solids, *SPIE's 45th Annual Conf., 4117 (Vision Geometry IX)* San Diego, 30 July - 4 August 2000, 100–111.

[3] F. Sloboda and B. Zatko, On approximation of Jordan Surfaces in 3D, *Technical report,Institute of Control Theory and Robotics*, Slovak Academy of Sciences, Bratislava, (2001)

[4] R. Klette, V. Kovalevsky, and B. Yip, On the length estimation of digital curves, *SPIE Volume 3811, Vision Geometry VIII*, Denver, 1999, 52–63.

[5] H. Minkowski, Volumen and Oberfläche, *Math. Ann.*, Leipzig, 1903, 447–495.