Fu-Che Wu · Wan-Chun Ma · Rung-Huei Liang · Bing-Yu Chen · Ming Ouhyoung

Domain Connected Graph: the Skeleton of a Closed 3D Shape for Animation

Abstract In previous research, three main approaches have been employed to solve the skeleton extraction problem: Medial axis transform (MAT), generalized potential field and decomposition based methods. These three approaches have been formulated using three different concepts, namely surface variation, inside energy distribution, and the connectivity of parts. By combining the above mentioned concepts, this paper creates a concise structure to represent the control skeleton of an arbitrary object.

First, an algorithm is proposed to detect the end, connection and joint points of an arbitrary 3D object. These three points comprise the skeleton, and are the most important to consider when describing it. In order to maintain the stability of the point extraction algorithm, a prong-feature detection technique and a level iso-surfaces function based on repulsive force field was employed. A neighborhood relationship inherited from the surface able to describe the connection relationship of these positions was then defined. Based on this relationship, the skeleton was finally constructed and named *Domain Connected Graph* (DCG). The DCG not only preserves the topology information of a 3D object, but is also less sensitive than MAT to the perturbation of shapes. Moreover, from the results of complicated 3D models, consisting of thousands of polygons, it is evident that the DCG conforms to human perception.

Fu-Che Wu · Wan-Chun Ma
Dept. Of Computer Science and Information Engineering, National Taiwan University, No.1 Roosevelt Rd. Sec. 4, Taipei, 106, Taiwan.
E-mail: {joyce,firebird}@cmlab.csie.ntu.edu.tw

Rung-Huei Liang
Dept. Computer and Communication Engineering, Ming Chuan University, No. 250 Chung Shan North Rd. Sec. 5,Taipei,111, Taiwan.
E-mail: liang@mcu.edu.tw

Bing-Yu Chen
Dept. of Information Management and Graduate Institute of Networking and Multimedia, National Taiwan University, No.1 Roosevelt Rd. Sec. 4, Taipei, 106, Taiwan.
E-mail: robin@ntu.edu.tw

Ming Ouhyoung
Dept. of Computer Science and Information Engineering and Graduate Institute of Networking and Multimedia, National Taiwan University, No.1 Roosevelt Rd. Sec. 4, Taipei, 106, Taiwan.
E-mail: ming@csie.ntu.edu.tw

1 Introduction

There are many methods to represent a 3D shape: Boundary representations (b-rep), Constructive Solid Geometry (CSG), implicit surfaces, and skeletal surfaces, as well as volume-based, point-based, and parameterization-based approaches. According to Pizer et al.'s suggestion [34], the skeletal surface method is a good way to represent shape because there are many applications directly related to a skeleton, such as object matching [18], animation [43], collision detection, and mesh editing [26]. In general, the skeleton-based method embeds local information and conforms more directly to human perception than other shape descriptions, and therefore it is suitable for handling a significant component as a unit. Because of its concise properties, the skeleton-based method is more efficient than other representations in computation [34].

However, the requirements for a skeleton differ with each application. For example, object matching may require skeletons to preserve principal features such as morphology (e.g. branching and termination) and geometrical information. Therefore, these skeletons can be regarded as major keys and then used to find similarities among objects [18]. On the other hand, object reconstruction requires a kind of skeleton that can reconstruct the complete geometrical information of the original surface [16]. The requirements must address the specific needs of various applications. When comparing almost identical shapes with only minor differences, it is acceptable to have the same skeleton; however, for surface representations, two different shapes require two different skeletons.

Although skeleton related applications have attracted a great deal of attention, it is not easy to extract a skeleton from an arbitrary shape that is at once robust, meaningful and manageable. Extracting a skeleton from a 3D object is

more difficult than from a 2D object [46]. Moreover, when handling different kinds of source models, such as non-manifold models, models which contain holes, or models which consist of many parts, many problems are encountered. Since providing this method for all kinds of models is not the major focus of this research, a pre-process procedure was used to convert the input models into 2-manifold meshes without boundaries, consisting of anywhere from thousands to tens of thousands of polygons.

The MAT is made from maximal contact spheres (or disks), found both inside and outside the object or the outline. A MAT consists of many spheres with different radii, used to encode an object. Consequently, if the curvature of the object surface varies, the MAT needs to describe this variation precisely to produce numerous links for its skeleton. By definition, MAT is sensitive to the variation of boundaries, or noise on its boundaries [10]. As Amenta et al. [1] go into in more detail; small perturbations in an object's boundary can induce large features in the medial axis. Some parts, however, of the medial axis are unstable, while other parts, corresponding to significant object features impervious to noise, are quite stable. A MAT skeleton consists of medial surfaces, curves and lines. It is a complex structure that is hard to handle as a control skeleton. The purpose of this research is to provide a skeleton that is suitable for further manipulation, such as skeleton-driven deformation [9]. In the opinion of the authors, certain properties must first be taken into consideration:

- Simplicity
  Because the extracted skeleton is desired for used in animation as a control skeleton, it must be simple and easy to control.
- Stability
  Two similar shapes should have two similar skeletons. If the shape of two objects only varies slightly, their skeletal structures should remain almost exactly the same.
- Meaningfulness
  The skeleton will be simplified to consist of only major prong parts without completely representing the details of a shape, such as width, surface noises, etc. Based on Kimia's paper [22], the medial axis plays a major role within the human visual system. Therefore, a skeleton which conform to human perception is preferred.
- Neutrality
  Because the authors want to bind the surface to the skeleton so that it can be distorted in any direction, the skeleton needed to be located at the medial axis. Hence, it is less biased around its surrounding surface.
- Hierarchical
  Each part in the skeleton has a different priority. According to the priority of each node in the skeleton, the skeleton's hierarchical properties can be applied to the problems of progressive refinement in transmission or of level-of-details (LODs).

The main idea of DCG is to represent a skeleton with a few linked points: a minimal number of points are selected to cover the whole surface, which is similar to the process of decomposition, where each part is represented as a meaningful unit in an object. Based on this idea, a main part is represented by a joint point, a prong part is modeled by an end point, and a variation on the skeleton is described by a connection point. This approach, illustrated in Fig.1, is an intuitive concept, and has been explored in the past [5, 39, 3, 24]. As well, the DCG skeletons of primitive models are shown in Fig. 2.
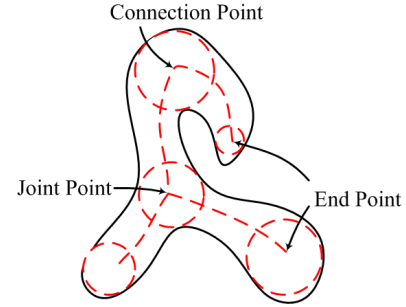

Fig. 1 Three main point types in a DCG skeleton: joint point, end point, and connection point.
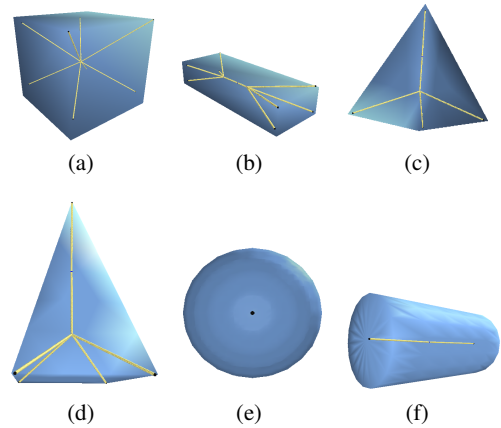

Fig. 2 The DCG skeleton of primitive models: (a)Cube (b)Brick (c)Tetrahedron (d)Pyramid (e)Sphere (f)Cylinder.

2 Related Work

Previous work regarding skeleton extraction can be divided into three categories: MAT, generalized potential field, and decomposition based approaches. These methods define different types of skeletons, and have been combined to construct a novel representation that avoids their respective disadvantages. In an abstract sense, MAT focuses on shape variation. Therefore, it is suitable for depicting the details of the

surface, and can be used to construct a preliminary model at the initial stage.

Since MAT is the basic definition of a skeleton, other methods also share the features of MAT. The potential field method focuses on the volume of an object, so it can be utilized to evaluate where a skeleton point would be suitable. The decomposition process can divide an object into many parts connected via linked points, and thus, a small number of meaningful connection points can be chosen to describe the variation of the skeleton.

2.1 Medial Axis Transform

The MAT based skeleton is a kind of reduced representation, and provides region-based shape features. Since its symmetrical properties are derived from its surface, it is suitable for reconstructing the original shape [44].

It is common to use Voronoi diagrams to construct the MAT skeleton of a 2D shape [29, 32]. Since the skeleton extracted by a Voronoi diagram contains many undesirable links, methods for pruning and reducing bifurcation have been proposed to solve these problems [7, 12, 27]. Choi and Seidel [11] have suggested a MAT pruning method whereby the one-sided Hausdorff distance of MAT is bounded, with respect to the boundary perturbation. Ogniewicz [31] proposes a residual function to measure the significance of skeleton edges. Attali and Montanvert [2] have shown that noises on the boundary can be modeled by a parameter graph, suggesting that noisy skeleton branches can possibly be characterized by a small maximal separation angle or thickness.

Sheehy et al. [36] have proposed an algorithm based on the domain Delaunay triangulation of a relatively sparse distribution of points, in order to construct the medial surface. After employing the Delaunay triangulation process, an adaptive refinement strategy is adopted to assemble the topology of the medial surface using a traversal of the classified tetrahedra in the triangulation.

Amenta et al. [1] have proposed the "power crust" algorithm for MAT approximation and 3D surface reconstruction. First, a Voronoi diagram is constructed. Then, a subset of Voronoi vertices are selected as poles. A weighted Voronoi diagram is then constructed from these poles and named the power diagram. These poles are subsequently labeled either inside or outside. The faces of the power diagram separating the cells of inside and outside poles constitute the power crust. The regular triangulation faces connecting the inside poles make up the power shape.

Tam and Heidrich [41] have proposed an approach for MAT that removes unwanted artifacts while preserving fine features. Their method prunes the axis by using an intuitive global threshold base on noise size, and then automatically reconstructs the fine features by extending the remaining branches.

Leymarie et. al. [24] have proposed that the shock scaffold is, in fact, a hierarchical and summarized organization of the 3D medial axis, in the form of a graph representation.

The shock scaffold is represented as a directed graph, where the medial points are the nodes in the structure, and the medial curves the links. The medial sheets are then represented by hyperlinks.

2.2 Generalized Potential Field

Another skeleton extraction approach is to use a distance field. In order to construct a distance field and extract the skeleton, voxel-based methods are frequently used. The basic methodology consists of constructing the distance field of an arbitrary object and then finding the local maximums of said distance field. The skeleton is then generated by connecting these local maximums. A skeleton based on these approaches is generally simpler than ones generated using MAT, since MAT based approaches usually include a thinning or shrinking process. They focus on where the most important part inside an object is, and in doing so, ignore the shape variation. Thus, the "thinning" is done at the cost of approximation, resulting in a non-unique skeletal representation. The problem of how to determine the point that indicates the end of a skeleton is, however, one which needs to be solved.

In 2D cases, Leymarie and Levine [25] implemented a 2D MAT (or Grassfire Transformation) by minimizing the distance field energy of an active contour model, the initial position of which is at the boundary of a given shape. Grigorishin et al. [17] have proposed using an electrostatic field to depict the inside energy, where select corner points are chosen on the boundary as starting points, used to track the skeleton path. Zhu [48] has proposed a stochastic algorithm to compute a skeleton in two phases. The first phase is to find some line segments in favor of parallelism and symmetry; the second phase of the process computes the axes and junctions. Siddiqi et al. [39] proposes a shock graph, whereby structural descriptions are derived from the singularities of a curved evolution process which acts on bounding contours.

As well, Borgefors et al. [8] have proposed a volume thinning approach when computing a skeleton for voxel-based approaches. This skeletonization method is performed in two major steps. The first step reduces the object to a surface skeleton. The second reduces the surface skeleton to a curve skeleton. Zhou and Toga [47] have proposed a voxel-coding method, which performs a recursive voxel-by-voxel propagation, and then generates the skeleton by using the Euclid and geodesic distance fields of an object. Bitter et al. [4] have proposed extracting a skeleton using volumetric data generated by a penalized-distance algorithm. Based on the distance field, the algorithm finds the furthest voxels as the centerline which in turn forms the initial skeleton. The skeleton is then refined recursively by discarding voxels near the centerline. Wade and Parent [43] also present an automated algorithm for generating a control skeleton based on voxel data, in which they use a thinning process, generated from the voxel data, to get their preliminary result. First, a central point is selected from this result as a root point, and

then, the furthest point is found iteratively to represent the end point. Finally, a control skeleton is formed by constructing a link between this point and the root point.

In addition to utilizing voxel data, some approaches which focus on the extraction of a skeleton directly from a boundary also exist. Sherbrooke et al. [37] and Culver et al. [14] have proposed an approach that first finds a seam and designates it to be the starting point, then searches for the path of the medial axis. Chung et al. [13] assume that each face of a polygonal object produces a force field; each point on the boundary is pushed by this force field which finally converges to a balanced position. This position is then labeled as a potential skeleton. In the end, all of these potential skeletons are connected to produce the main skeleton. Siddiqi et al. [38] propose the Hamilton-Jacobi skeleton method which can obtain the internal medial axis of an object, given its bounding curve or surface, by simulating the grassfire flow as a Hamilton-Jacobi equation using the Euclidean distance function. Li et al. [26] directly simplified the object mesh to get the skeleton. Based on the idea of edge contraction, their process iterates until all triangles have collapsed into edges or vertices. In other words, they are left with only skeletal edges, whose union is the skeleton of the mesh.

The previous work of the authors [28] proposed a thinning-like process that uses a radial basis function as the distance function in order to thin the surface of the object surface of the object, thus forming the skeleton. The results shown in Fig. 3 are similar to the approach currently being taken. However, the previous approach was not stable in some cases such as the 'donut-like' model, since there is no guarantee of constructing a radial basis function that keeps the monotonically increasing property well inside an object.
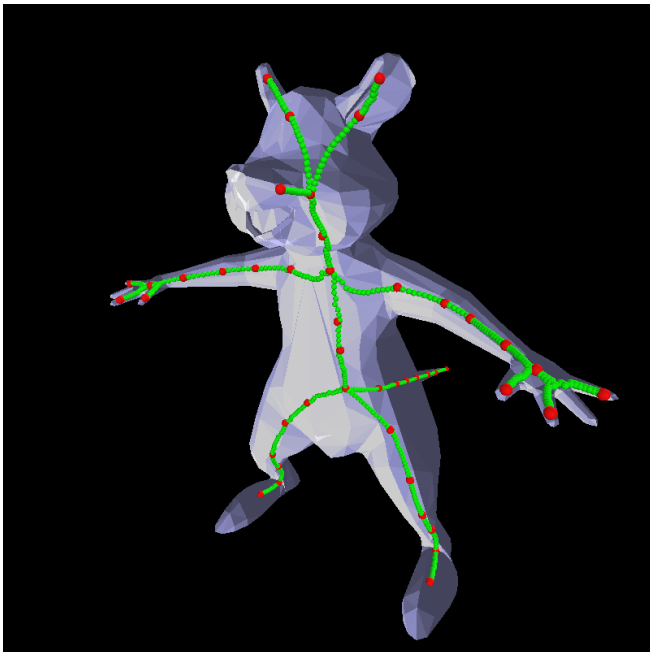

Fig. 3 Previous results based on radial basis functions.

2.3 Decomposition Based Method

This type of method has received much attention recently; it extracts a one-dimensional skeleton based on the Reeb graph. The idea of the Reeb graph is to use a continuous function, usually a height function, to describe the topological structure and to reveal the topological changes, such as merging or splitting. Based on a continuous function, the target object can be divided into many parts. Connecting the center position of all parts forms the main skeleton [40]. However, it can easily be seen that using a height function to construct the Reeb graph does not guarantee the result to be Euclidean-invariant, which is very important in shape analysis.

Since different functions may generate different Reeb graphs, Lazarus and Verroust [23,42] replaced the height function with the Geodesic Distance Function which is calculated by the Dijkstra algorithm; the generated skeleton is called a "Level Set Diagram" (LSD). Unfortunately, LSD depends on the selection of its source points. This results in the possibility of different LSDs being produced from the same model. Mortara and Patané [30] chose source points in high curvature regions to guarantee results that are Euclidean-invariant. Hilaga et al. [18] proposed a technique called "topology matching" that compares the Multiresolutional Reeb Graphs (MRGs) of 3D objects, in order to evaluate their similarities.

Katz and Tal [20] introduce an angular distance to measure shape variation, in order to identify a meaningful part. This approach, however, is not suitable for a 'donut-like' shape. Biasotti [3] proposed creating an extended Reeb Graph (ERG) by using a slicing strategy from a height map. With the extraction and classification of influence zones of critical points, the set of nodes of the ERG is completely defined. Each node codes the label of the critical area and its boundary components; thus, it is an abstract topological description.

3 Overview

The goal of this research is to find a concise skeleton that is suitable for editing the model and generating an animation object. To avoid the complex structure of a MAT skeleton, the skeleton was designed to be composed of a set of important points with linkages. There are three main stages for constructing a control skeleton.

– **Feature detection:** This initial procedure detects where the skeletal points of a 3D model can be suitably represented.
– **Connection construction:**
  To preserve the topology relationship of the original mesh, the neighborhood relationship of each feature point is inherited from surface edge connectivity.
– **Re-mapping for shape variation:**
  The variation of shape cannot suitable be described by two feature points connected by a straight line. There-

fore, a repulsive force field is introduced to force the skeletal path to follow the shape's varied form.

When discriminating a representative or small feature in 3D shape, it is not easy to find an absolute criterion. Based on experiments, a prong-feature detection algorithm was developed to discern where the signification features are. All other features in the shape, assumed to be noise, were ignored. Next, a skeleton structure was constructed to connect these signification features, thereby forming the main/principal skeleton.

To select the most important feature points as the domain points, a procedure was developed to evaluate which parts constitute suitable end features, connection features, and joint features. Fig. 4 shows different stages in the algorithm. The input model is shown in Fig. 4(a). A Voronoi diagram was then constructed to locate initialized candidates (Fig. 4 (b)). Domain balls were first extracted (Fig. 4 (c)). An end feature is locally far from the main part: a geodesic distance is measured on the surface, and then a watershed algorithm is applied to detect where an "end feature" is located (Fig. 4 (d))[45]. Then, a visibility repulsive force field is introduced, and all other features except the "end features" are adjusted to the local minima that constitute "connection points" (Fig. 4 (e)). Following this, a neighborhood relationship from mesh connectivity is determined (Fig. 4 (f)); a point that has more than two connection linkages is a labeled as a "joint point". Finally, a snake algorithm based on the repulsive force field is applied, and all linkages are adjusted to fit the shape variation (Fig. 4 (g)).

In the following excerpt, the repulsive force field will first be introduced. Based on this force field, the position where a skeleton is suitable to be located can be evaluated. Since an accurate computation of this field is time consuming, in order to reduce the complexity of the task, only a few candidates are selected. In the initial stage, the Vonoroi diagram is introduced to locate these candidates. Next, a prong-feature detection algorithm is presented. Finally, the method of constructing the connection to assemble the final skeleton is described.

4 Repulsive Force Field

Since it is necessary to identify the most important part inside a shape, a generalized potential field based on the level iso-surface function is introduced. The concept of the level iso-surface function is shown in Fig. 5.

Assume that shape S is defined on space Ω. A iso-surface function f based on the shape S can partition the space into $f(x) < 0$, $f(x) = 0$, and $f(x) > 0$ to represent outside the shape, the boundary of the shape, and inside the shape, respectively [35]. Since we only concern the level property of the potential field inside the shape, the function is defined as following. A level iso-surface function is a distance function on a spaced Ω. If such function $f : R^3 \to R$ is defined on the space Ω, all points on the boundary have maximal values. Moreover, a location near the boundary has a higher
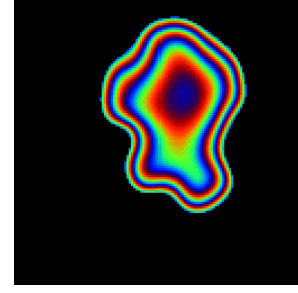

Fig. 5 The level iso-surface function.

value. For example, let d be the inverse of a shortest distance function, it satisfies the following properties:

1. If $p \in S$, $\forall q \in \overset{\circ}{S} \to d(p) > d(q)$, where $\overset{\circ}{S}$ is the interior of S.
2. If p and q are two points inside a shape S, $\|p\| < \|q\| \to d(p) > d(q)$, where $\|p\|$ is the shortest distance from position p to the boundary.

The balanced positions of the function d are the candidates of MAT, which consist of some faces and lines. Since the shortest distance is determined based on only one position on the boundary, the result is unstable and sensitive to noise. We want to construct a function that has only one global minimal point in a convex shape. Thus, a repulsive force field is defined as $F(x) = \int \frac{\overrightarrow{\mathbf{r}}}{r^n} d\theta$, where r is the distance from x to the boundary in the direction $\overrightarrow{\mathbf{r}}$, and a discrete form is

$$\mathbf{F}(p) = \sum_{0 \le \alpha < 2\pi; 0 \le \beta < \pi} \frac{\overrightarrow{\mathbf{u}}_{\alpha,\beta}}{\|p\|^n},$$

where $\overrightarrow{\mathbf{u}}_{\alpha,\beta}$ is a unit sample ray, and α and β are angles in the spherical coordinate. A level iso-surface function $f(x)$ can be defined as $\|\mathbf{F}(x)\|$. When $n \to \infty$, the repulsive force field is dominated by the shortest distance, and thus $f(x) \propto d(x)$. In some sense, n is a term for a smoothing effect. Each position receives more boundary influence for a smaller n. As shown in Fig. 6, the inner level of the iso-surface function is smoother with a smaller n. In our implementation, we adopt $n = 1$.
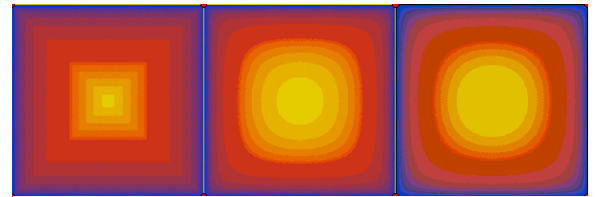

Fig. 6 The different n effects on the level iso-surface function for $n = 100$, 1, and 0.01, respectively.
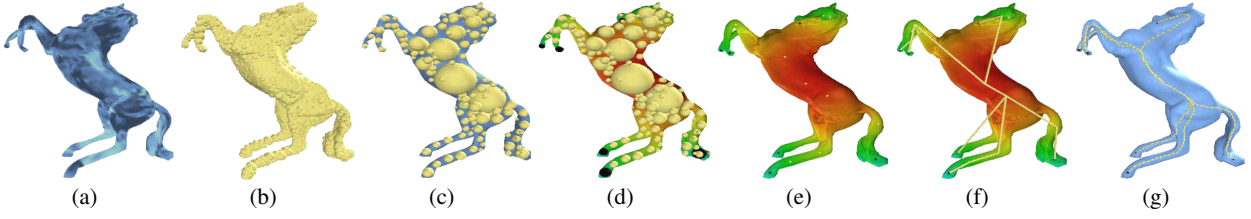
Fig. 4 Different stages in skeleton generation algorithm.

5 Initial stage

In this section, some internal points on the skeleton which represent or correspond to a wide surface portion are sought out. Points on the MAT are good starting points, since these points lie on symmetry axis of the shape. It is intuitive to locate the medial axis by using the Voronoi diagram [1]. The regions separated by the Voronoi diagrams are called Voronoi cells. For 2D points the cells are convex polygons; for 3D points, cells are convex polyhedra. Each face of a cell is equidistant between two points in a discrete set. Thus, a set of points on these faces lie roughly along the medial axis [19]. Bradshaw and O'Sullivan [6] have presented a sphere-trees construction algorithm to perform efficient collision handling. The first part of the algorithm ensures that a set of medial spheres cover the object's surface. The second phase of the adaptive algorithm is to improve the current approximation by adding more sample points. Based on Bradshaw's implementation, these Voronoi nodes are selected to be initial candidates. For constructing the Voronoi diagram, a set of sample points are constructed on an object's surface. After constructing the Voronoi diagram, each node finds its shortest distance to the boundary as its radius to form a Voronoi ball.

However, the node directly extracted from the Voronoi diagram left in this form remains too complex. For the purpose of stability, the effects of small variations on the shape need to be removed. There are many previous works that address this problem in 2D [11, 15]. Based on their ideas, the solution can be divided into two main concepts: one is to ignore small variations on the surface; the other is to find the principal data from the noise of the preliminary results. Using the former concept, Choi [11] developed a one-sided stability theory which enlarges the range of coverage of each skeletal point. In this way, small variations on the shape can be ignored. In the latter concept, Foskey [15] defines a θ-simplified medial axis based on the separation angle to extract the principal part from the MAT.

In our experiences, a detailed skeleton composed of many points is usually affected by noise or by small variations of the surface; on the other hand a rough skeleton (composed by few points) can be too loose to represent significant features. To avoid the noise effect, in this approach, only the important feature points in the first stage were kept. Then other necessary points were iteratively recovered, in order to preserve the variation of the original shape. Good feature points were located at the axial position and the noise effect on the surface was ignored.

Since larger radius means more range of coverage, a priority sequence of Voronoi balls under shape S can be defined, and each Voronoi ball labelled by the following numerical sequence: Let B_i is a Voronoi ball, its radius is ρ_i. If $\rho_1 > \rho_2 > \cdots > \rho_n > \cdots$, then $B_1, B_2, \cdots B_n, \cdots$ is a domain sequence.

Definition 1 (The Domain Ball) Let $B_1, B_2, \cdots B_n, \cdots$ be a domain sequence. The domain ball can be defined by inserting a Voronoi ball iteratively, based on sequential priority, and retaining its non-intersecting status. $B_D = \{B_i \mid \bigcap_{i=1}^{n} B_i = \emptyset$, where n is the maximal number $\}$.

After calculating the radii of all the Voronoi balls, some were chosen iteratively, based on the magnitude of their radii. Once a maximal ball is chosen, it will delete all neighboring balls that could possibly touch it. In order to decrease the number of balls, a minimal distance constraint between each ball was also introduced. Fig. 7 shows the result of the domain balls. Although not a unique a representation of the shape, calculations can be based on these initial candidates to obtain a final result.
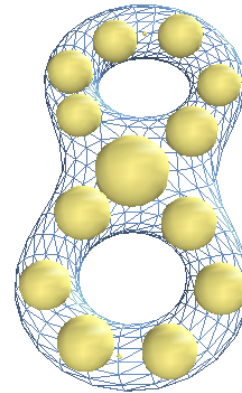

Fig. 7 The domain balls for a 3D model in the shape of the digit eight.

6 Prong-feature detection

Since a prong-feature may indicate the existence of a main skeleton, a prong-features detection algorithm is presented [45]. First, a normalized scale function is constructed on the input mesh to describe the position of a potential prong. The scale function is then calculated using the sum of the geodesic distance from each vertex among the mesh to the evaluated vertex [23]. To ensure that the calculation of the geodesic distance approximates the real value, a remesh pre-process is required.The scale function is defined in the following equation:

$$f(v) = \sum_{\forall v_i \in S} Geodesic(v, v_i),$$

where v is the vertex on the shape S. The results are as follows (Fig.8):

Fig. 8 From the scale function, a higher value indicates a prong feature denoted by the color green.

In order to find these prong features, a watershed algorithm is applied. First, all vertices are sorted by scale function value. Then, the maximum value of the vertex is picked up as the first prong feature and marked as traversed. As feature detection must be focused upon, the water level decreases iteratively from a maximum value. To understand how the watershed algorithm is applied to the mesh data with greater ease, the mesh data is represented in a parameterization form to describe the mapping relationship between the 3D and 2D cases. As viewed in Figure 9, the left side illustrates a model after calculating its scale function based on its geodesic distance. The right side shows a parameterization:

The mapping of the 3D mesh into a 2D map. The black line indicates the mesh on the 3D model. From this image, five green dots can be seen. These are highest values among the mesh, and also are the prong features to be detected.
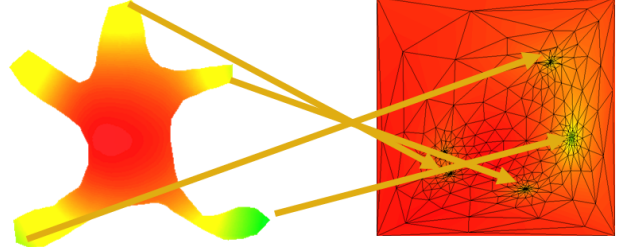

Fig. 9 The mesh data in a parameterization form.

In Figure 10, decreasing water levels are shown, in order to detect the local maximum. The function value is represented by the height. There are five local maximums which must be detected. The green mask represents the water level, which iteratively decreases. The new prong feature will be detected if it is not connected with the traversed region. Initially, the water level is setup to start at the highest value and then to decrease incrementally. If the water level decreases rapidly, as could possibly happen in this case, two prong features will become one. Thus, the decreasing levels of water affect the sensitivity of a prong feature. The prong shape's valley value, divided by its peak value, is defined as the smoothness ratio. If the smoothness ratio is larger than the threshold, the result will be a small prong, and will be ignored. Each non-traversed vertex in the sorting sequence will reset the water level, ie: its function value is multiplied by the smoothness ratio. Consequently, all vertices can be marked as traversed if their function value is above the water level and if they are a neighbor of the traversed set. If the vertex is not in the traversed set, it is a new prong feature. Thus, the nearest domain point to a prong feature is an **end point**.
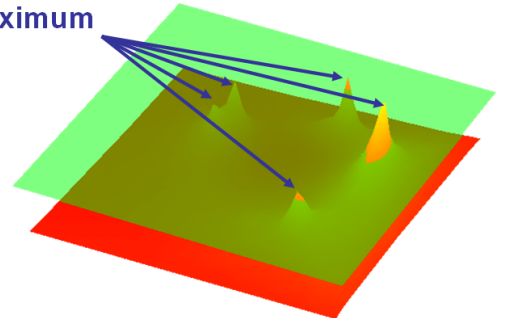

Fig. 10 The local maximum of a continue function is detected by the watershed algorithm.

An algorithm has been presented in order to detect the prong-features on a 3D mesh. The procedure includes a ratio for smoothness which can be adjusted to suit different applications. According to the experiences of the authors, for most models, setting the smoothness ratio to 0.85 was found to be suitable for detecting significant prong features.
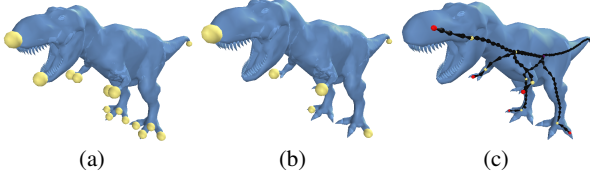

(a)   (b)   (c)

Fig. 11 The results are shown in different decreasing levels. (a) Fifteen features are detected when the smoothness ratio is set to 1.0 (b) Six features are detected when the smoothness ratio is set to 0.85. (c) A skeleton is constructed based on these prong features

7 Connection construction

In the previous section, the domain balls and end points were constructed. All domain points except end points are candidates in need of refinement. The refining process is based on the repulsive force field. For each point p_i, we use the repulsive force to push it to a new position p_{i+1}:

$$p_{i+1} = p_i + \mathbf{F}(p_i) \times step$$

The iterative process stops when the position reaches a local minimum $\|\mathbf{F}(p_i)\| < \|\mathbf{F}(p_{i+1})\|$. To describe the shrinking process, figure 12(a) shows that a point under the repulsive force field moves from the boundary and shrinks into the local minima. The shrinking paths are represented by the red lines. Because the shape contains two areas, there are two local minima inside which are represented by the blue circles. Due to the numerical inaccuracy of the step, not all the points will stop at an exact position. In practice, all intermediate points will converge to a location within a small range, which depends on the iterative step. To solve this problem simply, without substantial loss, these final positions are merged in a given range to obtain a single domain point, which is regarded as a **connection point**. Fig. 16(b) shows the shrinking process. However, a connection point is an intermediate point to help recover the whole topological information. Each point represents a local part.

A joint feature potentially connects more than two skeletal linkages. The topology relationship needs to be recovered to shows how things are connected. To recover the connectivity of domain points, the relationship between surface and domain points first needs to be determined. Thus, based on the connectivity of surface, the connectivity of domain points can be induced.

Definition 2 (Face Connectivity) Connectivity is a neighborhood relation denoted as $\underset{c}{\Leftrightarrow}$. If two faces share an edge,
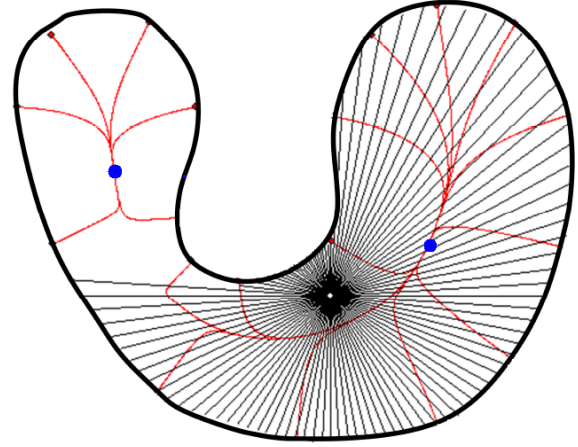
(a)
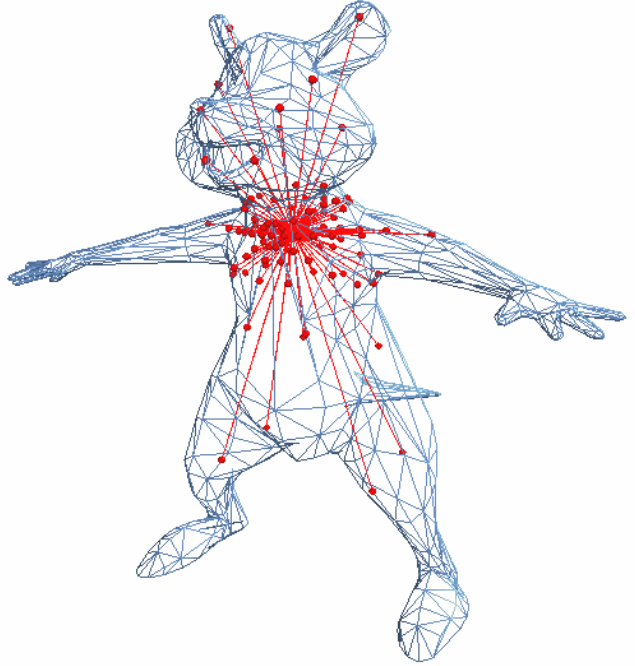
(b)

Fig. 12 A point receives the repulsive force from the boundary. (a),(b) are 2D and 3D cases, respectively.

the two faces have connectivity. Let a face A be bounded with some boundary edges ∂A. Thus, for two faces A_i, A_j, if $\partial A_i \cap \partial A_j \neq \emptyset \rightarrow A_i \underset{c}{\Leftrightarrow} A_j$.

Definition 3 (Domain Surface) *DS* is defined as a surface region which belongs to a domain point *DP*. Each domain point has a domain surface which can be described by this point. A domain point can find at least one shortest distance of point on the surface. A face that contains this point is called a primitive visible face denoted as A_{pv}.
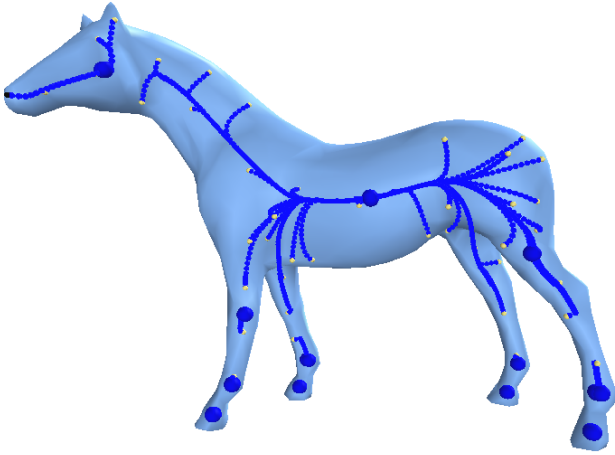
Fig. 13 The domain points are shrunk under the repulsive force field to form the connection points(larger blue balls).

1. $A_{pv} \in DS(p)$
2. Let $A_i \underset{c}{\Leftrightarrow} A_j$ and $A_j \in DS(p)$,then, $A_i \in DS(p)$ if and only if $\forall q \in DP, \|A_i p\| < \|A_i q\|$

Definition 4 (Domain Point Connectivity) Let $DS(p), DS(q)$ represented as domain surface of domain point p, q , respectively. If $\exists A_i \underset{c}{\Leftrightarrow} A_j, A_i \in DS(p), A_j \in DS(q) \rightarrow p \underset{c}{\Leftrightarrow} q$.
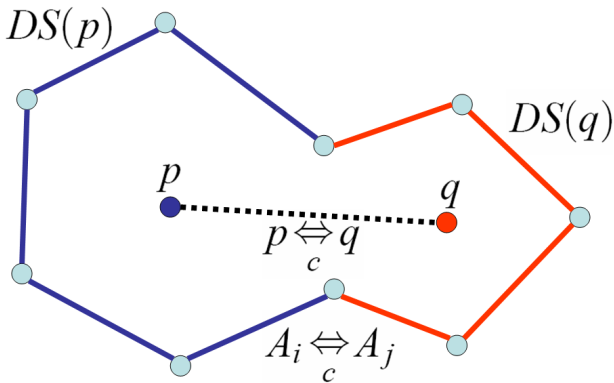

Fig. 14 Two domain points p, q are connected by the connectivity of the domain surfaces $DS(p), DS(q)$.

This idea is shown in Figure 14. Based on previous definitions, the algorithm can be implemented as follows. First, what domain point each face on the shape belongs needs to be determined. The touching position on the shape of each domain ball in was discovered during a previous process. Faces containing this position are primitive visible faces, so they belong to this domain point. Then, for the other unsolved faces, all domain points will be searched to find the

shortest distance of domain point that must be visible as their potential target. If a face potentially belongs to a domain point and its neighbor also belongs to the same point, the face is tagged with this domain point.

Finally, there are faces that are not visible to any domain points. These faces are defined by their neighboring faces, using the same domain point iteratively. If all faces have been defined with their domain points, edges shared by different domain faces will have a connectivity relationship between their domain points. The two domain points are connected via a connection link. A connection link is constructed, using many skeletal points as control points for a curve. A connectivity relationship constructed directly from the surface and a domain point may contain different parts of domain surface on opposite sides, such as in a cube case. This leads to connectivity problems. Besides under these circumstances, this situation happened in a joint location. Fig 15(a) shows it is possible for a cycle linkage to be formed. A path cost is evaluated by using the mean force under the repulsive force field. A larger force of linkage is then deleted, because it is far from the main skeleton. If this linkage is construed from a ring on the surface, this is a genus case, and the cycle will be abandoned. This type of case can be determined by observing the following condition: If each point in the cycle is visible among other points, it can not be a genus case. Fig. 15(b) shows these results.
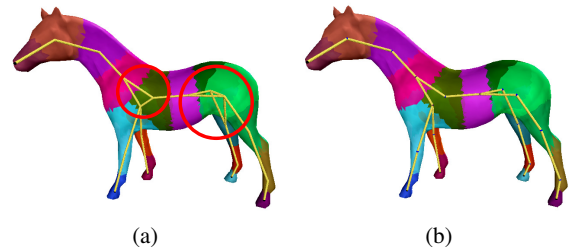

Fig. 15 A cycle linkage is deleted based on the mean force of the linkage under the repulsive force field.

From the previous definition, the connectivity of the domain point is inherited from the connectivity of the domain surface. After constructing the connectivity relationship among domain points, a connectivity counting function $C(DP)$ can be defined to report how many connectivity relationships each domain point containing. The **joint point** is defined as $J = \{DP | C(DP) > 2\}$

Now the geodesic distance function, end points, and joint points have been found. The re-mapping process will be described in greater detail in the following section.

8 Re-mapping

To describe the variation of the shape faithfully, a re-mapping procedure on these skeletal points among a linkage is re-
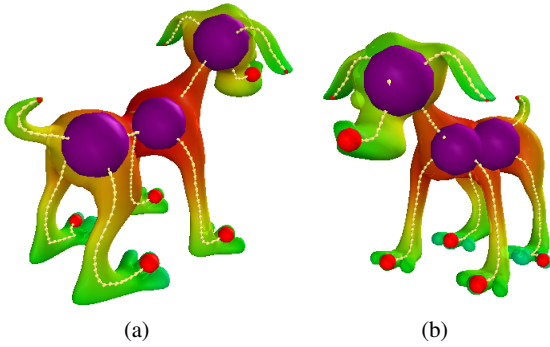
Fig. 16 The model is represented by the geodesic distance function. A purple ball indicates a joint point. A red ball indicates an end point.

Fig. 17 The deformation process of a link, where the green points between two red points (end point and joint point) are pushed by the generalized force field to the final position.

quired. Additionally, such a fix must be used to ensure the links are within the object's field.

Lemma 1 (Skeletal Point) *Let $P = \{x | \mathbf{n} \cdot x = c\}$ be a cutting plane, where $\mathbf{n}$ is the normal, c is the distance from the origin to the plane. A cutting plane can divide an object into two parts. Each part contains one domain point. Thus, the cutting plane contains at least one skeletal point. $SP = \{p | p = f_{min}(p)$, $\mathbf{n} \cdot p = c\}$, where f_{min} is an evaluation function to determine where the placement of a skeletal point is suitable on the cutting plane*

For two parts to be connected in a shape there must be a skeletal point to connect the two parts. The location of a skeletal point is determined by its local shape attributes, such as scale and connectivity [33]. The scale relates to geometry to obtain correct skeletonization. The connectivity relates to the topology, and concerns itself with how things are connected. If two domain points have a connectivity relationship, there is a link between these two points. Initial points can be generated to represent this link. To begin with, selected points are sampled along a straight line to represent a connection link. Here, a Snake [21] algorithm (Active Contour Model) is invoked to adjust these points so that they correspond to a skeletal curve. An active contour model is generated by adjusting a contour to minimize an energy function:

$$E_{Snake} = E_{Internal} + E_{External},$$

where $E_{Internal}$ and $E_{External}$ are internal and external energy respectively. Internal energy is the part that depends on the intrinsic properties of the link, such as length or curvature. $E_{External}$ means external energy, and its value depends on the generalized potential field. The position of each point is adjusted on the constrained plane, which is perpendicular to the original straight line. When the energy of the active contour model is minimized, the result skeleton is obtained. Fig. 17 shows the deformation process of a connection link.
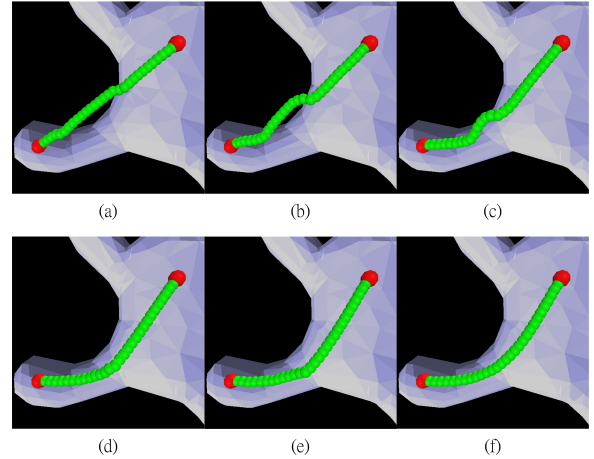
9 Results

Fig. 19 shows the DCG of nine typical 3D models, and Table 1 shows the execution times for the different stages. The timing statistics were obtained using an Intel Pentium-M 1 GHz processor notebook with a 512 Mbytes memory. In the time statistics, the watershed algorithm is not listed, because in this step, its operator is only to check the function value of each vertex, and thus, its computation time is less than a second. Currently, the re-mapping procedure consumes the most CPU time, and the working process takes has not yet been optimized.

Our algorithm includes four main parts, which are as follows. (1) Compute the Voronoi diagram (2) Geodesic distance on the mesh (3) Watershed on the surface (end feature detection) (4) Snake algorithm on 3D potential field. For a n vertices of a model, the computation complexity of algorithm (1), (2) and (3) are $O(n^2)$. Thus, this process is not feasible for a large model. The complexity of the snake algorithm is not easy to determine from the number of the vertex. It is based on the shape variation and the number of skeletal points. However, for efficiency and convergency, we limit the iterative loop of the optimization process to no more than one hundred times. The result is acceptable. To avoid long computation times, the original models have been simplified first into any number of polygons, from one to ten thousand. Fig. 20,21,22,23 shows more results from. Their skeleton data in Maya format are available at:
http://www.cmlab.csie.ntu.edu.tw/cml/g/DCG/.

Further applications of Maya are also demonstrated by using the current results to produce an animation sequence. Fig. 18 shows this sequence.

As has been suggested previously, there are properties to evaluate what a good skeleton is. The result of this research has determined that most cases correspond with these properties. In MAT, the skeleton consists of faces and lines. The skeleton generated in this project has only line struc-

Model	Face number	Voronoi diagram (min.:sec.)	Geodesic distance (min.:sec.)	Connection construction (min.:sec.)	Re-mapping (min.:sec.)
dragon	7315	00:56	00:13	01:43	05:16
eight	1536	00:06	00:05	00:12	00:54
octopus	8648	01:06	00:45	05:28	13:01
dinosaur	5605	00:33	00:22	01:29	03:24
bunny	6343	00:27	00:14	06:14	15:01
hand	9936	01:33	00:25	03:04	27:01
rhino	7678	01:06	00:26	02:09	07:43
cheetah	5345	00:33	00:15	01:14	03:19
rabbit	2174	00:11	00:10	00:24	02:27

Table 1 Execution time statistics for nine 3D models on a Pentium-M 1 GHz PC
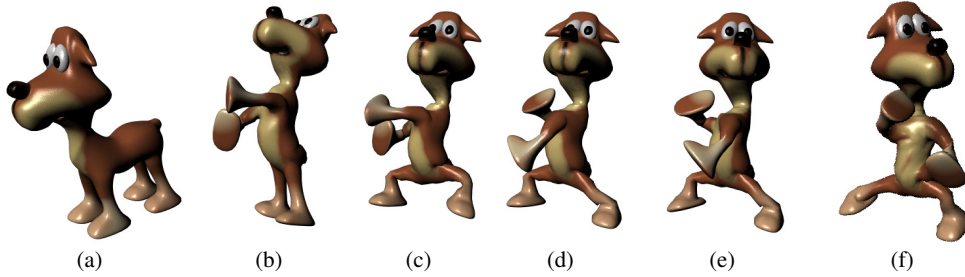
.

Fig. 18 Using the project's results, an animation sequence is produced from Maya.

ture. Thus, it is simpler than MAT, and therefore suitable for skeleton-driven deformation. In addition, because the skeleton generated for this project is constructed using only signification features, with small variations on shape being ignored, it is inherently more stable. The resulting skeleton is acceptable in appearance for most people. Since a snake algorithm is involved under the repulsive force field, the skeleton has been re-mapped into a neutral location inside the 3D shape. Thus, the deformation of the skeleton is less biased for the binding shape. Regarding hieratical issues, based on the smoothness ratio, different importance of skeletons can be found. Because the skeleton generated in this project is based on the detected prong-features on the shape, it has been re-mapped into a neutral location under the repulsive force field. Thus, it is simple, stable, meaningful, and fair for shape binding and can, in addition, be hieratical.

In general, the MAT is better for surface reconstruction. The authors' skeleton produces a 1D graph that is better to control and is suitable for direct conversion to skeleton based applications such as Maya. A Reeb graph based skeleton is a decomposition based method. In it, a model has been deconstructed into many parts. Then, the skeleton is constructed by connecting the central point of each part. However, the location of the joint point is not well defined, which could possibly be unfair to the other side of the binding surface. The DCG method varies from others' in three crucial ways: Prong-features are considered to be end points, there are less joint points, and the skeleton paths are smoother.

10 Future Work and Applications

The definition of the end point, joint point, and connection points of a skeleton are proposed in this paper. Based on the definitions we proposed, model skeletons can also be automatically constructed. However, since the main concern of this project is to extract the principal part of a shape, some smaller parts or details of the shape will be ignored. To solve this problem, the shape can be iteratively refined by decomposing the object into many parts [20]. A small feature within the whole object becomes a big feature when a sub-part is scaled into a unit. The decomposition process cuts an object with a cutting plane. On the cutting plane, a connection point can be found in order to connect the two parts. Thus, a decomposed part must have a connection point on its shape to connect to the original skeleton.

Based on current results, each meaningful part is extracted and represented as a concise form. This research is suitable for many applications. In animation, each domain point is an easily controllable point. In object matching, the resulting skeleton could be a primary key for comparing the similarity of 3D objects [18]. However, it is only suitable to compare the similarity of the skeletal structure, and is not for extracting the exact object. In mesh editing, each domain surface could be handled as a unit. In mesh compression, residual between domain point and its described surface is small and the connectivity relationship is simple, making efficient compression very possible. Furthermore, the level of detail and a progressive display are also possible, based on the result skeleton of this research. In some applications, if a very concise form is not needed, connecting the domain

balls at the first stage will quickly generate a prototype. Finally, for symmetrical shapes such as hands and legs, because their skeletons are calculated individually, they may not be exactly symmetrical. The symmetrical properties of a skeleton are, however, an issue worthy of exploration in the future.

11 Conclusions

In this paper a concise representation for the skeleton of an arbitrary shape is proposed. The nodes of the graph are named domain points (significant points inside the shape) which can represent the local properties of a shape. The domain points are classified into three categories: joint points, end points, and connection points. In this paper, the concept and the definition of the DCG is introduced. Moreover, using an algorithm to generate the graph has been proposed, and as well, the differences between the DCG skeleton and the traditional MAT skeleton are described, explaining the advantages of the DCG. Given an arbitrary 3D shape, the proposed skeletonization method can extract both topological and geometrical information. The resulting skeleton can converge in limited computational time and is completely inside the boundary of a 3D shape. This is a concise, stable and meaningful representation of a general 3D object can avoid the problems of noise-sensitivity encountered in the MAT. Furthermore, there are fewer restrictions on these types of 3D models, a problem often faced by the Radial Basis Function based approaches.

References

1. Amenta, N., Choi, S., Kolluri, R.: The power crust. In: Proc. SM 2001, pp.249-260 (2001)
2. Attali, D., Montanvert, A.: Modeling noise for a better simplification of skeletons. In: Proc. IEEICIP 1996 **3**, pp.13-16 (1996)
3. Biasotti, S., Falcidieno, B., Spagnuolo, M.: Surface shape understanding based on extended reeb graphs. In: Surface Topological Data Structures: An Introduction for Geographical Information Science, S. Rana and J. Wood, Eds., John Wiley&Sons, pp.87-103 (2004)
4. Bitter, I., Kaufman, A.E., Sato, M.: Penalized-distance volumetric skeleton algorithm. IEEE TVCG 7(3), pp.195-206 (2001)
5. Blum, H.: A Transformation for Extracting New Descriptors of Shape. MIT Press, pp.362-380 (1967)

6. Bradshaw, G., O'Sullivan C.: Adaptive medial-axis approximation for sphere-tree construction. ACM TOG 23(1), pp.1-26 (2004)
7. Borgefors, G., Nyström, I.: Efficient shape representation by minimizing the set of centres of maximal discs / spheres. Pattern Recognition Letters 18, pp.465-472 (1997)
8. Borgefors, G., Nyström, I., di Baja, G.S.: Computing skeletons in three dimensions. Pattern Recognition 32, pp.1225-1236 (1999)
9. Capell, S., Green, S., Curless, B., Duchamp, T., Popović, Z.: Interactive skeleton-driven dynamic deformations. ACM TOG 21(3), pp.586-593 (2002). (Proc. SIGGRAPH 2002)
10. Choi, H.I., Choi, S.W., Moon, H.P.: Mathematical Theory of Medial Axis Transform. Pacific Journal of Mathematics. **181**(1) pp.57-87 (1997)
11. Choi,S.W., Seidel, H. P.: One-Sided Stability of Medial Axis Transform, Lecture Notes in Computer Science, **2191**, pp.132-139 (2001)
12. Choi, W.-P., Lam., K.-M., Siu, W.-C.: Extraction of the Euclidean Skeleton based on a Connectivity Criterion, Pattern Recognition, **36**, pp.721-729 (2003)
13. Chung, J. -H., Tsai, C.-H., Ko, M.-C.: Skeletonization of Three-Dimensional Object Using Generalized Potential Field, IEEE Transaction on Pattern Analysis and Machine Intelligence, **22** (11), pp.1241-1251(2000)
14. Culver, T., Keyser, J., Manocha, D.: Accurate Computation of the Medial Axis of a Polyhedron, Proceedings of ACM symposium on Solid Modeling and Applications, pp.179-190 (1999)
15. Foskey, M., Lin, M.C., Manocha, D.: Efficient Computation of A Simplified Medial Axis, Proceedings of ACM Symposium on Solid Modeling and Applications, pp.96-107(2003)
16. Giblin, P., Kimia, B. B.: A Formal Classification of 3D M.A. Points and their Local Geometry, IEEE Transaction on Pattern Analysis and Machine Intelligence, **26** (2), pp.238-251(2004)
17. Grigorishin, T., Abdel-Hamid, G., Yang, Y. H.: Skeletonization: An Electrostatic Field-Based Approach, Pattern Analysis and Applications, **1**, pp.163-177(1998)
18. Hilaga, M., Shinagawa, Y., Kohmura, T., Kunii T. L.: Topology Matching for Fully Automatic Similarity Estimation of 3D Shapes, SIGGRAPH 2001 Conference Proceedings, pp.203-212(2001)
19. Hubbard, P.M.: Approximating polyhedra with spheres for time-critical collision detectio, ACM Transactions on Graphics, **15** (3), pp.179-210(1996)
20. Katz, S., Tal, A.: Hierarchical Mesh Decomposition using Fuzzy Clustering and Cuts, SIGGRAPH 2003 Conference Proceedings, pp.954-961(2003)
21. Kass, M., Witkin ,A., Terzopoulos, D. : Snakes: Active Contour Models, International Journal of Computer Vision, **1**, pp.321-331(1987)
22. Kimia, B. B.: On the Role of Medial Geometry in Human Vision, Journal of Physiology-Paris, **97**(2-3), pp.155-190(2003)
23. Kimia, F., Verroust, A.: Level set diagrams of polyhedral objects, In ACM Symposium on Solid Modeling and Applications, pp.130-140(1999)
24. Leymarie, F. F., Kimia, B., Giblin, B., Towards, P. J.: Surface Regularization via Medial Axis Transitions, International Conference on Pattern Recognition, pp.123-126(2004)
25. Leymarie, F. F., Levine, M.D. : Simulating the Grassfire Transform Using an Active Contour Model, IEEE Transaction on Pattern Analysis and Machine Intelligence, **14** (1), pp.56-75 (1992)
26. Li, X. -T., Woon, T.-W., Tan, T.-S. , Huang, Z.-Y.: Decomposing Polygon Meshes for Interactive Applications, Proceedings of ACM Symposium on Interactive 3D Graphics, pp.35-42(2001)
27. Nilsson, F., Danielsson, P.-E. : Finding the Minimal Set of Maximum Disks for Binary Objects, Graphical Models and Image Processing, **59**(1), pp.55-60 (1997)
28. Ma, W.-C., Wu, F.-C., Ouhyoung M.: Skeleton Extraction of 3D Objects with Radial Basis Function, Proceedings of Shape Modelling International 2003, pp.207-215(2003)
29. Mayya, N., Rajan, V. T.: Voronoi Diagrams of Polygons: A Framework for Shape Representation, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp.638-643(1994)

---

[1] http://www.3dcafe.com

30. Mortara, M., Patane, G.: Affine-Invariant Skeleton of 3D Shapes, Proceedings of Shape Modeling International 2002, pp.245-278(2002)

31. Ogniewicz, R.: Automatic Medial Axis Pruning by Mapping Characteristics of Boundaries Evolving Under the Euclidean Geometric Heat Flow onto Voronoi Skeletons, Technical Report 95-4, Harvard Robotics Laboratory(1995)

32. Ogniewicz, R., Ilg, M.: Voronoi Skeletons: Theory and Applications, Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp.63-69(1992)

33. Palenichka, R. M., Zaremba, M. B.: Multi-scale Model-Based Skeletonization of Object Shapes Using Self-Organizing Maps, Intl. Conference on Pattern Recognition 2002, pp.10143-10147(2002)

34. Pizer, S. M., Thall, A. L., Chen ,D. T. Chen : M-Reps: A New Object Representation for Graphics, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp.638-643(1994)

35. Savchenko, V. V., Pasko, A. A., Okunev, O. G., Kunii T. L.: Function representation of solids reconstructed from scattered surface points and contours, Computer Graphics Forum.,**14**(4), pp.181-188(1995)

36. Sheehy, D. J., Armstrong, C. G., Robinson ,D. J. ,: Shape description by medial axis construction, IEEE Transactions on Visualization and Computer Graphics,**2**(1), pp.62-72(1996)

37. Sherbrooke, E. C., Sherbrooke, Patrikalakis, N. M., Brisson E. : Computation of the Medial Axis Transform of 3D Polyhedra, Proceedings of ACM Symposium on Solid Modeling and Applications, pp.187-199(1995)

38. Siddiqi, K., Bouix, S., Tannenbaum, A. R., Zucker, S. W.: Hamilton-Jacobi Skeletons, International Journal of Computer Vision, **48**(3), pp.215-231(2002)

39. Siddiqi, K., Shokoufandeh, A., Dickinson, S. J., Zucker, S. W.: Shock Graphs and Shape Matching, International Conference on Computer Vision, pp.222-229(1998)

40. Shinagawa, Y., Kunii, T. L.: Constructing a Reeb Graph Automatically from Cross Sections, IEEE Computer Graphics and Applications, pp.44-51(1991)

41. Tam, R., Heidrich, W.: Feature-Preserving Medial Axis Noise Removal, ECCV2002, pp.672-686 (2002)

42. Verroust, A., Lazarus, F.: Extracting Skeletal Curves from 3D Scattered Data, The Visual Computer, **16**(1), pp.15-25(2000)

43. Wade, L., Parent, R. E.: Automated Generation of Control Skeletons for Use in Animation, The Visual Computer, **18**(2), pp.97-110 (2002)

44. Wolter, F. -E.: Cut Locus and Medial Axis in Global Shape Interrogation and Representation, Technical Report(1993)

45. Wu, F.C., Chen, B.Y., Liang, R. H., Ouhyoung, M.: Prong Features Detection of a 3D Model Based on the Watershed Algorithm, ACM SIGGRAPH2004 Sketches (2004)

46. Wyvill, G., Handley, C.: The Thermodynamics of Shape, Proceedings of Shape Modeling International 2001, pp.2-8. (2001)

47. Zhou, Y., Toga, A.: Efficient Skeletonization of Volumetric Objects, IEEE Transaction on Visualization and Computer Graphics, **5** (3), pp.195-206(1999)

48. Zhu, S.-C.: Stochastic Jump-diffusion Process for Computing Medial Axes in Markov Random Fields, IEEE Transaction on Pattern Analysis and Machine Intelligence, **21**(11), pp. 1158-1169(1999)
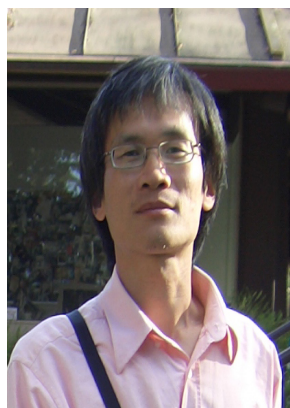
Fu-Che Wu received the B.S. degree in Mechanical Engineering from the Feng Chia University, Taichung, in 1987 and received the M.S. degree in Computer Science and Information Engineering from the National Chiao Tung University, Hsinchu, in 1996. He is actually working towards a Computer Science and Information Engineering Ph.D. in the National Taiwan University, Taipei. His research interests are mainly in Shape Analysis, Geometric Modeling, and Computer Animation, particularly related to skeleton extraction.

Wan-Chun Ma received his BS degree in computer science and information engineering from National Taiwan University in Taipei, Taiwan. He is currently a PhD student in the Communication and Multimedia Laboratory of the Department of Computer Science and Information Engineering at the National Taiwan University. His research interests include shape analysis, skeleton extraction, reflectance, and real-time rendering.

Rung-Huei Liang is an assistant professor of Dept. Computer and Communication Engineering at Ming Chuan University. His research interests include facial/gesture recognition and virtual reality applications. He received the B.S. and Ph. D. degrees in computer science from National Taiwan University. He is a member of ACM SIGGRAPH.

Bing-Yu Chen received the B.S. and M.S. degrees in Computer Science and Information Engineering from the National Taiwan University, Taipei, in 1995 and 1997, respectively, and received the Ph.D. degree in Information Science from the University of Tokyo, Japan, in 2003. He is currently an assistant professor in the Department of Information Management and the Graduate Institute of Networking and Multimedia of the National Taiwan University since 2003. His research interest are mainly for Computer Graphics, Geometric Modeling, Web and Mobile Graphics. He is a member of IICM, ACM, and IEEE.

Ming Ouhyoung is a professor in the Graduate Institute of Networking and Multimedia and the Department of Computer Science and Information Engineering, National Taiwan University. His research interests include computer graphics, virtual reality and multimedia systems. He received a BS and a MS in Electrical Engineering from National Taiwan University, and a Ph.D from the University of North Carolina at Chapel Hill. He is a member of IEEE and ACM.
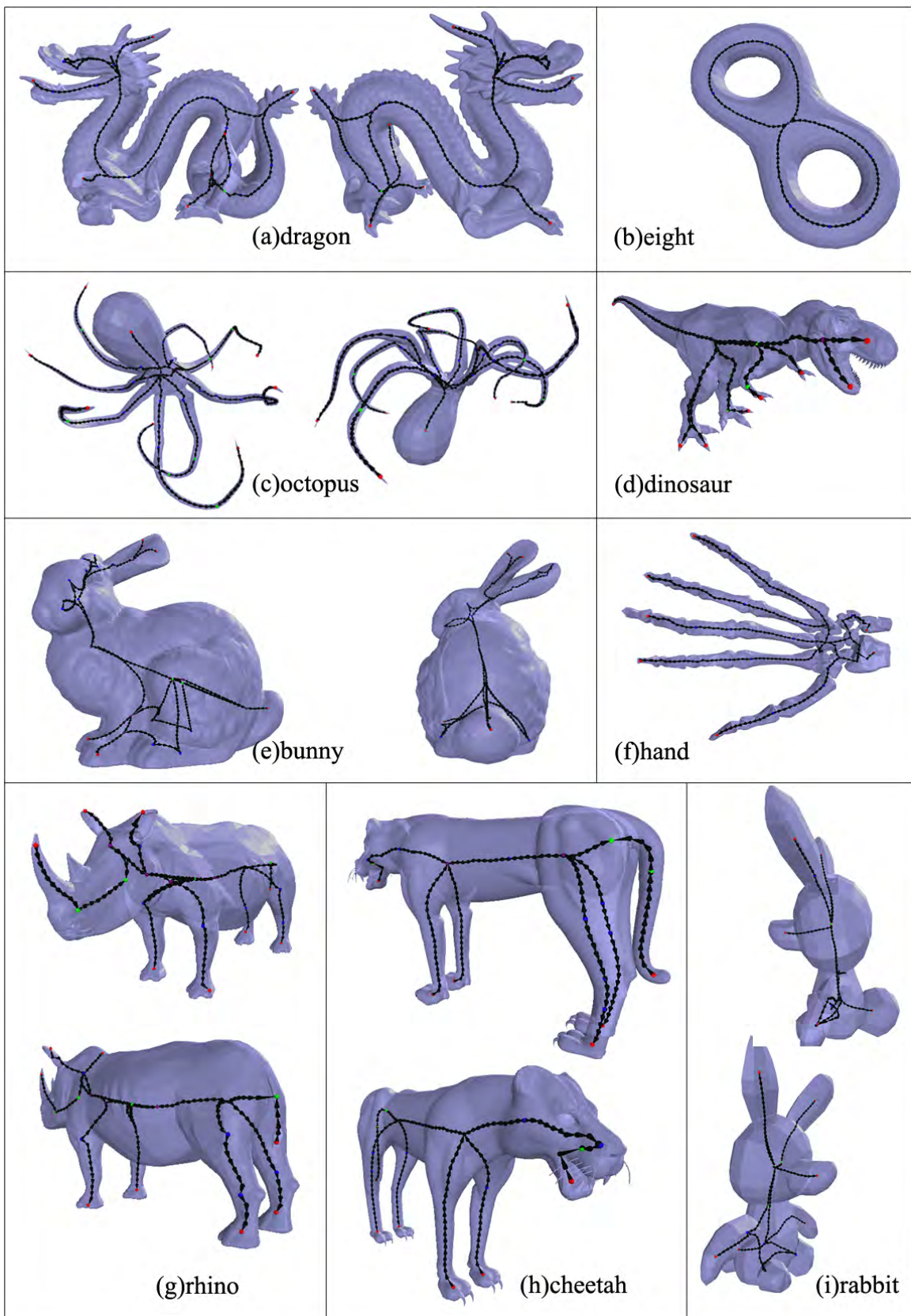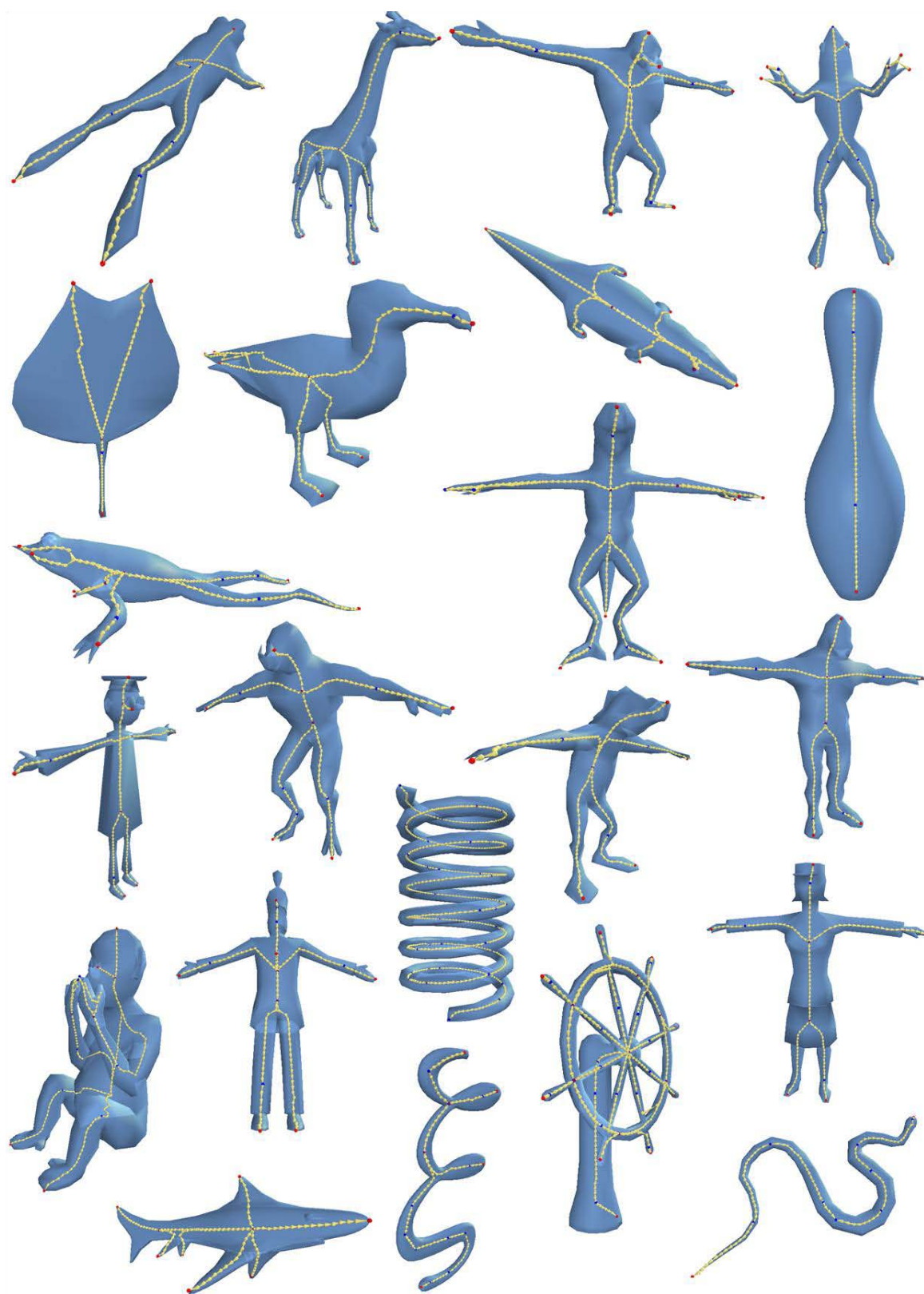
Fig. 19 There are results from different kinds of models. For performance consideration, the models have been simplified firstly. The number of faces are under ten thousand.

Fig. 20 More results with skeleton data are available at **http://www.cmlab.csie.ntu.edu.tw/cml/g/DCG/**.

Fig. 21 Results

Fig. 22 Results

Fig. 23 Results