We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,300

Open access books available

172,000

International authors and editors

190M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

Volume Decomposition and Hierarchical Skeletonization for Shape Analysis

Xiaopeng Zhang¹, Bo Xiang¹, Wujun Che¹ and Marc Jaeger²

¹LIAMA-NLPR, Institute of Automation, CAS
China

²CIRAD-AMAP / INRIA-Saclay
France

1. Introduction

Volume is one of the important shape types in our world, from daily tools to complex and precise equipments, and even to life phenomena. Shape analysis techniques help understanding and using shape information for various applications. With the development of shape data acquisition and digitalisation techniques, more and more high-resolution shape data sets are available. Increasing demand for their compact shape description in applications inspires the need to reduce the data to a description of more concise remnant. Skeletonization and volume decomposition are fundamental tools for shape information processing and understanding, being widely used in many applications, such as character animation, measurement and navigation planning in virtual colonoscopy.

A skeleton is an ideal shape representation with significant data compression while highlighting topological structures. The skeleton of a solid object, accompanied with radius information, exhibits its shape variation and spatial expansion. Skeletonization is a process of data abstraction to extract skeletons of an object, being promising and efficient due to linearity and simplicity of skeletons. There is an extensive body of scientific literature on 2D Skeletonization. 3D skeletonization is widely cared about in recent years. It has been linked to different shape-related techniques, like shape manipulation (Katz & Tal, 2003), shape matching (Funkhouser et al., 2004), shape retrieval (Tung & Schmitt, 2004) and collision detection (Li et al., 2001).

Shape decomposition is a practical process of dividing complex structures of an object into simple components. It has been indicated that good shape decomposition can result in skeleton extraction of high quality (Katz & Tal, 2003), and that a high quality skeletonization may lead to a meaningful decomposition (Li et al., 2001). But we will have much work to do yet in specifying the relationship between shape decomposition and skeletonization of 3D shapes (Lien & Amato, 2007). The definition of shape decomposition of a volume is still challenging since it is hard to specify the cut-surface.

2. Related Work

Volume elements, usually known as *voxels* in literature, are evenly distributed in three axis directions, constituting the space occupation of the volume. Skeletons and shape components are typical shape features. The skeleton of a volume is 1D thinning structure to represent its shape and topology (Brostow et al., 2004), and it is a concise representation of the volume shape. Shape decomposition is a basic technique to describe its complex structure. Related work on 3D skeletonization, shape decomposition and their applications are briefly discussed in this section.

2.1 Distance based volume skeletons

Distance transformation of a volume converts voxels into layers in terms of a feature point or the boundary surface. A skilful concept was proposed in (Zhou & Toga, 1999) to find a shortest path skeleton through the combination of DFS-distance (Distance From a Starting point) transformation with DFB-distance (Distance From Boundary) transformation. Fig. 1(a) shows a DFS-distance map and Fig. 1(b) shows a DFB-distance map. A global minimum cost path-searching algorithm was provided in (Bitter et al., 2000) with penalty distance through a heuristic combination of DFS-distance and DFB-distance. It was extended in (Sato et al., 2000) to branched skeletons with an adaptive sphere. The penalty distance algorithm was further improved in (Bitter et al., 2001) to correct mistakes caused in (Bitter et al., 2000) and (Sato et al., 2000).

A thinness parameter was adopted in (Gagvani & Silver, 1999) to control the candidacy of a voxel on a skeleton using DFB-distance transformation. A set of underlying skeleton points is defined by maximum central point map (MCP) (Fig. 1(c)).

The corner-cutting problem of volume skeletonization was solved in (Wan et al., 2002) by delivering a centred path rather than a shortest one with exact Euclidean distance and using minimum-spanning tree (Fig. 1(d)). But the influences of side-branches on the main ones are not considered.

Fig. 1. Distance based volume skeletons. (a) DFS-distance; (b) DFB-distance; (c) MCP; (d) Centreline of a colon

2.2 Thinning based volume skeletons

Parallel volume thinning is an efficient way to find the centreline by deleting outer voxels iteratively if the deletion does not destroy the original topology of the object.

Ma & Sonka (1996) proposed a fully parallel and connectivity-preserving thinning algorithm to reduce computational cost. 3D thinning erodes a 3D binary image layer by layer through templates to extract the skeletons. Expensive testing of feature points is avoided by

matching the 26-neighborhood of the points with predefined templates (Fig. 2(a)). However, the final centreline may be disconnected and many spurious branches may be generated in the resultant skeleton too. The connectivity of the skeleton is kept in (Liu et al., 2005) by adjusting a point deletion with its neighbour points in different iterations and a length parameter is also applied to removing the creation of spurious branches. However, these methods are ineffective if the length of a spurious branch is longer than that of a real one. Ma's Templates was modified to preserve connectivity in (Wang & Basu, 2007), as shown in Fig. 2(b), where the left is the original object, the middle is the skeleton with the approach of (Ma & Sonka, 1996) and the right is with that of (Wang & Basu, 2007).

Fig. 2. Thinning based skeletons. (a) Ma's templates. (b) Modified Ma's Templates

2.3 Shape decomposition and skeleton hierarchy

Shape decomposition and skeleton hierarchy are significant topics of many shape information processing, such as shape analysis and understanding of 2D models (Rom & Medioni, 1993; Simmons & Sequin, 1998), 2D images (Siddiqi & Kimia, 1995; Telea et al., 2004), boundary represented models (Au et al., 2008) and volumes data (Cornea et al., 2005). Lien & Amato (2004) used approximate convex decomposition (ACD) to partition the mesh into nearly convex components and skeletons are then extracted from their convex hulls, respectively. Au et al. (2008) presented a simple and robust skeleton extraction method based on mesh contraction. As shown in Fig. 3(a), the 1D skeleton shape is achieved by performing geometric contraction using constrained Laplacian smoothing.

A framework was presented in (Reniers & Telea, 2007) to segment a 3D shape into meaningful components using curve skeletons. Critical points, or ramification point, are used to construct a partition of the object surface with geodesics, and the segments have minimally-twisting smooth borders. The resultant segmentation of the shape reflects the hierarchical structure of the curve skeleton, as illustrated with Fig. 3(b). But the application of volume decomposition to hierarchical skeletons is not considered. The work of (Reniers & Telea, 2007) is based on an assumption that all skeletons compete equally around each ramification while our work proposed here un-equally.

Fig. 3. Shape decomposition and skeleton hierarchy; (b) Skeleton by Mesh Contraction; (a) Segmentations through curve skeletons

The concept of hierarchical curve-skeleton was proposed in (Cornea et al., 2005) and (Cornea & Min, 2007). A family of hierarchical curve-skeletons is extracted robustly for varied 3D objects, e.g. volumetric shapes, polygonal models or scattered point sets. These algorithms are based upon computing a repulsive force field over a discretization of the 3D object. Topological characteristics of the resulting vector field, such as critical points and critical curves, are used to extract the curve-skeletons.

2.4 Applications of shape features

2D shape decomposition was concerned for shape recognition before 3D case. In the work of (Siddiqi & Kimia, 1995), a partition scheme was proposed and used to segment a variety of 2D shapes. Different scales of partitions are intuitive by relating segmented parts to semantic portions of the original object, which are rather useful to shape recognition (Fig. 4(a)).

Fig. 4. Applications of shape features to shape retrieval and analysis; (a) Recognition of 2D shape through decomposition; (b) Mesh reconstruction through medial axis

Shape decomposition and skeleton hierarchy provide shape features of a volume for various applications. Ju et al. (2007) proposed a new method by alternating thinning and a novel skeleton pruning routine to extract skeletons of volumetric models for shape description. This technique is simple and meaningful, but cannot be used to segmentation, matching and recognition.

Hierarchical skeletons can be used for solid reconstruction. Local differential geometry of different kinds of points forming 3D symmetry sets was analysed in (Giblin & Kimia, 2004). With this approach, the extracted skeletons can be used to reconstruct the original surface with full geometrical information. Linked with a real-time simulation of stroke therapy, a mechanism was proposed in (Luboz et al., 2005) to segment and to reconstruct 3D human vasculature models with a balance of smoothness, number of triangles and distance error.

The idea of power crust presented in (Amenta et al., 2001) was used to approximate the medial axis transform (MAT) of an object. An inverse transform was then applied to producing mesh representation of the surface from MAT. Examples in Fig. 4(b) suggest the capability of surface reconstruction from laser range data.

Skeletonization of a range image from a real tree is a very new development. Xu et al. (2007) and Cheng et al. (2007) used range images to reconstruct the geometric model of a tree. The former is for producing full a polygonal model of a range scanned tree through skeletonization of the trunk and main branches of the tree (Fig. 5(a)), while the latter is through 2D skeletonization, cylinder fitting and generalized surface (Fig. 5(b)).

Fig. 5. Applications to tree reconstruction; (a) Knowledge and heuristic-based modelling of laser-scanned trees; (b) Tree Branch Reconstruction through cylinder fitting

2.5 Contributions of our work

A new approach of volume decomposition and efficient extraction of hierarchical skeletons are systematically described in this chapter based on our work as in (Zhang et al., 2008, 2009).

Five aspects are discussed: computing multiple distance transformations to find the hierarchical structure of the object volume; decomposing the volume into simple components; extracting compact and smooth skeletal segments corresponding to each independent components; efficient skeleton extraction from noise data of laser scan; and an application of this technique to plant reconstruction. Technical contributions are fourfolds: (1). The influence of side branches to the main one is avoided via volume decomposition; (2). The extracted hierarchical skeletons keep well the topology of the original object; (3). Skeleton nodes sampling is adjusted to smooth the skeletons; (4). The new approach is applied to handling the shape of more complex topology, e.g. with a loop.

The deviation problem in skeletonization is solved with volume decomposition, compact skeletonization and hierarchical skeletons. The relationship between shape decomposition and skeletonization is emphasized and specified with technical details. The construction of volume decomposition surfaces, or cut-surfaces, and skeleton point sampling are extended so that curve-skeletons become smoother and keeps well the shape of the branched volume

than our early work. The extraction is fully automatic and much faster, even to a shape with complex topology like a ring volume.

3. System Overview

In this chapter we process tree-like object volumes possibly with some loop structures. The main technical line is to decompose the volume into components, based on which hierarchical skeletons are extracted.

Volume decomposition is to separate a complex branched volume into a series of simple components around its ramification points, each of which is topologically equivalent to a single column. The term *hierarchical* in this chapter does not mean a hierarchical process for skeleton extraction but a hierarchical decomposition of a volume into components. Skeletons are extracted from all components and then are connected to form a hierarchical structure. The object data should have an evident root point, i.e. the centre at the bottom of the trunk.

Three main concepts are concerned as classification-skeleton for volume decomposition, decomposition for hierarchical connection of skeletons and path growing for efficient skeletonization. *Classification-skeleton* shows the hierarchical structure of the object volume, so it is chosen as the key criterion for volume decomposition. *Hierarchical connection* means a connection of skeletons into a tree-like structure in the same topology with the original volume after decomposition. Classification-skeleton is the standard for connection. *Path growing* means voxel propagation from a seed point in one direction until some conditions are satisfied.

The classification-skeleton is extracted through distance transformation from the boundary or a seed point and through central cluster graphing. With the help of classification-skeleton, volume decomposition is performed by cross-section surfaces, called cut-surfaces; each decomposition component should be close to the ramification points so that the deviating affect of side branches is greatly reduced.

Finally, the hierarchical branched skeleton is organized by connecting all compact skeleton segments according to the classification-skeleton so that the final hierarchical skeleton reflects the shape of the original volume.

In order to extract skeletons efficiently, distance transformations from multiple source points are made from all tip points so that all components of the object can compete to reach the root through ramifications. They are used for both decomposition and skeletonization. A cluster interval is adopted to generate equivalent skeleton point range for each component so that the skeleton becomes smoother and more concise.

The system includes four main aspects: (1) Voxel classification into a hierarchical structure through integer seed point distance transformation and boundary distance transformation, then classification-skeletons for branched structure; (2) Volume decomposition at ramification points through real-valued distance transformation based on classification-skeletons; (3) Compact skeletonization of each decomposed component through real-valued distance transformation and connection of hierarchical skeleton on reference of classification-skeletons; (4) Path competition for decomposition and efficient extraction of smooth skeletons. Fig. 6 details the contents and the structure of the system.

Fig. 6. Overview of the system

4. Voxel Classification

Classification of voxels according volume topological structure consists of five steps.

4.1 Distance transformation

Distance transformation is a tool for voxel classification. It is a propagation process of voxel values from a voxel sub-set S to the whole set V . At the beginning, the value of each voxel in V is initialized, usually as 0 or 1. We take W as the work set, and S as the initial values of W . In each step of propagation, all voxels in the n -neighbourhood of a voxel in W are set as m_n+m if its initial value is smaller than it, where m is the distance of its n -neighbourhood voxel in W and m_n is the scheme for distance transmission between two n -neighbouring voxels. Often the evaluation rule is represented as $m_6-m_{18}-m_{26}$, where m_6 , m_{18} and m_{26} are the evaluation transmissions to 6-neighborhoods, 18-neighborhoods and 26-neighborhoods respectively. They are selected according to desired transformation precision, such as 1-2-3, 4-5-6 and $1-\sqrt{2}-\sqrt{3}$.

m_n can be an integer, or more precisely, a real. Integer number is good for concise classification and real number is good for precise shape acquisition. In this section integer is adapted before volume decomposition,, such as 1-2-3 or 3-4-5, so that distance can be used

as a criterion to distinguish voxels according to branch structures. After volume decomposition, real is adapted instead for a precise skeleton.

Voxel points can be organized after integer distance transformation. A *cluster* is defined as a set of consecutive points with the same integer distance. *Classification skeleton* is a structure of skeletons linking the volume root to each branch tip, which shows the hierarchical structure of the object volume. *Volume decomposition* is the separation of a complex branched volume around its ramification points into a series of simple components, each of which is topologically equivalent to a single column. *Hierarchical connection* means a connection of skeletons into a branched structure of the same topology with the original volume.

Classification skeleton is a direct application of integer distance transformation. It is used to find out tip points, ramification points, hierarchical structure and a rough central skeleton. The skeleton obtained this way is not well central and not 26-neighboring either, but it can be used as a criterion for voxel classification due to its integer value.

4.2 Central clustering

Integer seed point transformation is performed with the evaluation scheme 1-2-3 and 6-neighborhood propagation, where the seed point is the volume root specified by the user or by the system. The main purpose of the transformation is to search all tip points of branches and all ramification points so that decomposition could be properly performed at each ramification point.

Fig. 7. Central clustering on root distance

An *S-cluster* is the set of all the points having the same *S-distance* value, which is similar in part to sphere waves from the seed point as its centre (Fig. 7(a)). *S-cluster* is often abbreviated as a *cluster* here. All clusters constitute a graph according to their neighbouring relation. The *positive direction* of this graph is defined as from the root to one tip.

A *tip cluster* is composed of points with local maximum *S-distance*, which corresponds to a volume tip. A *ramification cluster* is one whose oneness breaks up at the next neighbour clusters in the positive direction of the cluster graph. Seeing locally from a ramification cluster in the positive direction, the *S-distance* of the next neighbouring clusters will increase by one. A ramification cluster corresponds to a volume ramification. The tip cluster and ramification cluster represent the topological structure of the cluster graph.

A *transition cluster* is one that is not a seed cluster, not a tip cluster or not a ramification cluster. A transition cluster has only one preceding cluster and only one succeeding cluster. Fig. 7(b) shows the structure of all clusters, where circles represent tip points, and triangle represents the root point. After omitting all transition clusters, the cluster graph becomes a directed graph with each cluster as a node of the graph (Fig. 7(c)). This graph is referred to as a *cluster graph*.

4.3 Construction of the branch-link path

A *branch-link path* is a sequence of points with two ends, i.e. the root point and one tip point. All branches are correctly connected with branch-link paths. A branch-link path is unnecessarily the centreline of the volume, but it is the basic for finding a central path of the volume. It is calculated as follows.

All tip clusters are found first. For each tip cluster, the maximum B-distance point, or the barycentric point of all the maximum B-distance points if they are more than one, is chosen as the initial current point. For each current point, we search for the point in its 6-neighborhood with least S-distance as the next point, which is either in the same cluster or in its neighbour cluster. This process is repeated until the root seed point is reached; then a sequence of points is obtained, starting from a tip point to the root point with decreasing S-distances in turn.

Because the distance between two neighbour clusters is no more than 1, a branch-link path is guaranteed to converge to the root point in the decreasing order.

Since 6-neighborhood is used, on the other hand, there is always a point in the path with the distance of any integer number between 1 and the distance number from the tip point to the seed one. Therefore, this sequence contains all the clusters between the tip and the seed.

Only one representative of a cluster is reserved and the connection of all representatives from the root to the tips will become a bundle graph or a divergent graph. This graph is called the *branch-link path*. It is not a medial axis of the volume yet.

4.4 Centralization of the branch-link path

Centralization of a branch-link path means moving this path to the centreline or replacing each point of the path with another one in the same cluster closer to the cluster centre, so that the entire path approaches the medial axis of the object volume. Two concepts are useful for centralization. One is *Maximum central point set C* of all points in the maximum central point map. *Boundary-distance maximum point set B* is the point set with maximum boundary distance.

The centralization begins from the tip point of each branch in the direction of decreasing S-distance. If *C* in this cluster is not empty, its barycentre is accepted as a representative of the cluster; or else, the barycentre of *B* is selected; and the boundary distance is recorded as the corresponding radius. If both *C* and *B* are empty, the one with smallest distance to the immediately preceding point is chosen, and the boundary distance is the corresponding radius.

In the method of (Shahrokni et al., 2001), either the point with the maximum B-distance is chosen, or, if they are multiple, their barycentric point is chosen. However, since the connection relationship of neighbouring clusters is not well considered, the resultant central point may destroy the original topology of the volume. The green circle in Fig. 8(a) shows

this wrong node location and wrong topology. We exploit both C and B to centralize the branch-link path so that the final path is correct in centrality and topology (Fig. 8 (b)).

4.5 Main branch and branched structure

Fixing the main branch is to find a primary and secondary order of all branches around each node, then to select the principal path and to unite all of them around the node according to their mutual distances. The determination of filiation and brotherhood depends on the radius of each branch at the node. A simple way is to judge by the length, but it sometimes fails since the main branch is not always the longest, but more often the thickest.

After the determination of all paths and all ramification points, all filiations are recorded and all these relationships constitute the branched structure. Considering the computation errors due to integer distance transformation and the influence of side branches on the parent branch, this hierarchical skeleton is not ideal, but can be used to find the ramification location and volume decomposition at the ramification. It contains the information of hierarchical classification, centrality and connectivity, so it is called the *classification skeleton*.

5. Volume Decomposition and Compact Skeletons

5.1 Volume Decomposition

Volume decomposition means the separation of a branched volume around ramification points into a hierarchy of components, each of which is topologically equivalent to a single column. The crucial technique for decomposition is how to position cut-surfaces to divide the volume into parts.

The object volume is regarded as of clear hierarchical structure of parent-child relation at each ramification point. After separation, the main branch is called a *parent branch* or a *principal branch*, and the others are called a *subordinate branch*. All cut-surfaces are cross sections almost orthogonal to the skeleton direction. The basic consideration in constructing a cut-surface is to find a surface on each branch and to avoid any two cut-surfaces intersecting with each other. Additionally, a cut-surface is as close to a ramification as possible so that as many voxels as possible will be assigned to the branch from its parent. The search for cut-surfaces starts from tips to the root.

The general topology of the volume and hierarchical connection relationship of components can be deduced through the classification-skeleton obtained above.

For clarity, we only describe the construction algorithm for a bifurcating point. We take any skeleton node location as a seed point and perform an integer seed point transformation in the entire volume. This is the first stage of transformation for volume decomposition. For each integer distance value, there must be a cut-surface or some disconnected cross-section surfaces associated with this distance. The branched structure of the volume around a ramification is responsible for these surfaces, called *wave-pairs*. Each surface is similar to a part of a sphere, and it is almost orthogonal to the classification-skeleton. Each wave pair is chosen to be as far as possible from the ramification position. Then a pair of surface transformations is performed backward from the wave-surface-patch pair surfaces to the ramification location until these pair surfaces intersect with each other. This is the second stage of volume decomposition. The reason for the progress from farthest to nearest is to make the cut-surface as orthogonal to the skeleton as possible. Also, cut-surfaces too far from the ramification location will leave more voxels in the parent branch.

Fig. 8. Cluster graph and segmentation; (a) Improper cluster graph; (b) Proper cluster graph; (c) the seed point transformation with the ramification as the seed point; (d) a pair of wave clusters are selected from the seed point transformation; (e) the cut-surface

We set the cut-surface into the child branch, and it is called the root-surface of the child branch. Its centre is called the *main source point* of the child branch. The corresponding surface consists of voxels in the parent branch but 26-neighboring to some voxels of the root-surface of the child branch. It is called the side-surface of the parent branch.

Fig. 8 (c)-(e) depicts an example of volume decomposition at a ramification. Fig. 8 (c) is the seed point transformation of a trachea at the ramification. Fig. 8 (d) displays two separate cut-surfaces with the same distance from the seed to the ramification in two different branches. Fig. 8(e) illustrates the final result of the volume decomposition, green corresponding to the parent branch, blue corresponding to the filial branch, and red representing the cut-surface.

After decomposition, the shape of each component has several mouths, as shown in Fig. 9(a). The barycentric voxel of the main tip mouth of this branch component is called the *main tip*. The barycentric voxel of the mouth corresponding to its child volume is called a *side tip*. The barycentric voxel of the mouth cut from its parent branch or the root mouth is called a *main source*, which is the seed point for this component.

5.2 Compact Skeletons

The results of volume decomposition are used to extract skeleton segments of component V , to record the connection relation and to connect them with more skeleton segments into the hierarchical topology equivalent to that of the volume.

Our method of hierarchical skeleton extraction takes three steps: (a) generation of a 3-D directed weighted graph from each component data; (b) using a minimum spanning tree (MST-tree) to get a skeleton segment for each component; (c) connection of all skeleton segments. For each component, the main source point is chosen as the seed for point distance transformation. The evaluation scheme is real value $1 - \sqrt{2} - \sqrt{3}$ with 26-neighborhood propagation for better precision. Then, C -Cost function $C(p)$, i.e. a *Compact Centrality Cost Function*, is constructed. Finally, a branched structure is constructed of minimum central cost connecting the main source point and the main tip point, and this topological structure is equivalent to that of the volume before volume decomposition.

5.2.1 Compact centrality cost

The central line is the feature inverse to the boundary, so the distance to the boundary is chosen as the basic opposing element of Centrality. In order to have centrality and

compactness connection of the skeleton both be considered, $C(p)$ is defined as an addition of two parts in (1): boundary cost, B -cost $B(p)$, and compactness cost, Q -cost $Q(p)$.

$$C(p) = B(p) + Q(p) \quad (1)$$

where $B(p)$ is defined as $1/b(p)$, and $Q(p)$ as a scaled accumulated seed distance cost $T(p)$ in (2).

$$Q(p) = k(V) \xi(V) T(p) \quad (2)$$

where $k(V)$, $\xi(V)$, $T(p)$ are calculated by (3), (4) and (5) respectively.

$T(p)$ is defined in a recursive manner starting from the main source point in Fig. 9(a), where the main source point is chosen as the seed point for transformation, and its $T(p)$ is set as 0 at this point. Then the $T(p)$ is evaluated recursively in (3), like a seed distance cost expansion from the seed point.

$$T(p) = \min \{ S(q) + (1 - M * b(p)) E(p, q) \mid q \in N_n^*(p) \} \quad (3)$$

where $M = \min \{ B(p) \mid p \in V \}$ is the minimum cost in current component V . The effect of $b(p)$ is to make p closer to the centre and $1 - M * b(p) \in [0, 1]$. $E(p, q)$ is defined as $d(p, q) - 0.75$, ($q \in N_n^*(p)$), where $d(p, q)$ is the Euclidean distance. The effect of -0.75 is to make the influence of the distance between neighbouring voxels lesser. For any two neighbour voxels p and q , $d(p, q)$ is in $[1, \sqrt{3}]$, so $E(p, q)$ is in $[0.25, 0.919]$.

$\xi(V)$ in (2) is defined as the minimum absolute difference of any two B-distances of two voxel points in volume V in (4).

$$\xi(V) = \min \{ |b(p) - b(q)| : |b(p) - b(q)| \geq \nu; p, q \in V \} \quad (4)$$

Thus the Q -cost at any point must be smaller than difference between any two points in the volume. Considering that real-valued B -distance could make $\xi(V)$ very small, a double precision real number is used to represent this cost. On the other hand, we perform a simple transformation on all of the B -costs and they will be treated as equal when the difference of two B -costs is smaller than a small value ν . Therefore, the effect of $\xi(V)$ is that Q -cost will differentiate two points of the same B -cost; a voxel closer to the centre has a bigger Q -cost. For voxels of different B -cost, Q -cost will have no effect since they have no impact on the two different B -costs. Therefore, the selected point will be closer to the centre and closer to the seed point simultaneously so that centrality and compactness are both satisfied.

The coefficient $k(V)$ in (2) is a constant for component V and it is defined as (5).

$$k = 1/T, \quad T = \min \{ T(p) \mid p \in V \} \quad (5)$$

The centrality cost is the global cost of non-centrality and non-compactness.

5.2.2 Construction of a minimum spanning tree

After the determination of the compact centrality cost for each voxel, the volumetric dataset is converted into a 3D directed weighted graph. This graph includes all the links of any two 26-neighboring voxels. The connection directions of any two voxels are bi-directional, with different values for different directions. The weight is determined by the C-cost

Similar to (Wan et al., 2002), all component data are transformed to 3D directed weighted graphs at first, and then a minimum-spanning tree is constructed for each component. This is expressed in the following procedure:

- Step 1) Mark source point S as the seed point for C-cost. Mark S , set its parent pointer *parent-link* to NULL.
- Step 2) Select the seed point as the beginning current point T .
- Step 3) For each current point T , push each of its unmarked 26-neighborhood B_i into a sorted heap in increasing order of C-cost, so that the one with the smallest C-cost is always at the top. If the C-cost of the *parent-link* of the node B_i is larger than that at T , set the *parent-link* of B_i to T .
- Step 4) Pop the top node of the heap, mark it, and set it as the current node.
- Step 5) Repeat Step 3) and Step 4) until the heap is empty.

This is a little similar to the algorithm described in (Wan et al., 2002), but they did not consider ramifications. Our method addresses this issue, more concisely, compactly and robustly due to formula (3). The resultant skeleton always stays away from the boundary and the tendency to go roundabout is avoided.

5.2.3 Skeleton extraction and hierarchical connection

Skeleton extraction is in two steps. The main tip and side tips are calculated at first for the minimum-spanning tree of the current component (Fig. 9(a)). The skeleton segments with two ends of these two tips are called the *main-skeleton* and *side-skeleton*, respectively. The main-skeleton connects the main source and the main tip, whereas the side skeleton connects one side tip to the main-skeleton, so that the centrality of the skeleton, hierarchy, and topological connection are maintained.

Fig. 9. Hierarchical volume decomposition; (a) Volume structure after decomposition; (b) Airway 1, (c) Airway 2, (d) Larch tree, (e) Willow tree

Then the main-skeleton is extracted by tracing along the MST backward from the main tip point until the main source is met. The radius values of the skeleton segment are the

distances of the path points to corresponding boundary points. Skeleton extraction is finished after recording this path along with the associated radii. For the link-skeleton, the process is similar, except that the beginning point is chosen as the side tip point until one voxel in the main-skeleton is reached. From the definition of C-cost, the link-skeleton will converge to the main-skeleton since the latter is the skeleton of this component.

After the extraction of skeleton segments, i.e. main skeletons and link skeletons, they will be connected together according to their original topological relations, including all paths from each tip to root passing all nodes. This is where the hierarchical skeleton can help, with its excellent centrality and classification.

Main skeleton can be used for pattern recognition of 3D objects.

5.2.4 Experiments on volume decomposition and hierarchical skeletons

Table 1 lists the details about the data with compact skeletonization, including the number of voxels in the volume, the number of voxels in the skeleton, and the number of branches.

Fig. 9 (b)-(e) shows the results of volume decomposition, where green, blue, yellow, pink and cyan represent the levels of branches in increasing order. Objects here are two human airways, above year-old larch tree and a four year-old willow tree. It can be seen that each component has a single-column shape and their borders are kept well. Fig. 9 (b)-(e) also demonstrates that our volume decomposition technique is valid for multiple furcations.

Name	Volume Voxel No	Skeleton Voxel Number	Branch Number
Airway 1	552595	408	3
Airway 2	581539	1229	14
Larch	29657	910	12
Willow	49883	1622	18

Table 1: Volume and skeleton specifications with compact skeletonization

Fig. 10. Effect comparison among three approaches on three samples

The results of skeleton extraction by Ma's algorithm (Ma & Sonka, 1996), Wan's approach (Wan et al., 2002) and our algorithm are compared in Fig. 10. The skeletons of Ma's are shown in the first row, those of Wan's in the second row and those of our algorithm in the

third. The first column displays the results of the three algorithms at the ramification of an airway. It can be seen that spurious branches are pruned in our algorithm while they still exist in the other two. The second column illustrates their results with a simple ramification of a willow tree. Here we find that our algorithm can be used to eliminate the influence of deviation, but the other two fail. The third column is the skeleton of a willow branch. Ma's method generates some spurious branches in the ramification; Wan's generates a "roundabout" shape in the skeleton, but all these mistakes are removed in our algorithm. The fourth column illustrates the skeletons in a multi-ramification of the willow tree; our algorithm keeps the skeleton in the centre while the skeletons generated by the others do not follow the medial axis.

Overlaying of the final skeletons over the objects is illustrated in Fig. 11 from all four models in Fig. 9, where they are viewed from three angles. Non-boundary voxels are displayed with a non-transparent single colour model, and boundary voxels are with a semi-transparent and lighting model. The frontward boundary voxels are set semi-transparent, and rearward voxels are set to be non-transparent. The results demonstrate that skeletons extracted with our algorithm are more centred and more compact, and the deviation is also eliminated in the ramification part through compact centrality cost when there exist many candidate points with the same distance from the boundary.

Fig. 11. Overlaying skeletons on 3D models

6. Efficient Extraction of Skeletons

The work here on efficient extraction of skeletons is based on all the techniques described above. The extended work in (Xiang et al., 2008; Ma et al., 2008; Ma et al., 2009) makes skeleton extraction more efficient and more robust. The skeleton extraction method supports shapes with loop structures because each voxel in the volume, including those in the loop, can be reached in the path growing and labelled to a component of the volume, while it is not considered in (Zhang et al., 2008) with the minimum spanning tree.

DFS-distance transformation is applied to the volume data once. No skeleton refinement and connection are needed after the skeleton extraction so that make the method efficient.

6.1 Automatic detection of feature points

All feature features of the object volume are thought of as tip points,. The most important is the root point. Instead of specifying it by user as an input in (Zhou & Toga, 1999) and (Xiang et al., 2008), feature points are detected automatically in our work here.

A voxel v is chosen randomly in the volume as the seed to perform the DFS-distance transformation. A new feature point s is chosen as the centre of those voxels with the largest distance away from v . Point s is called the opposite feature point to v . s can be thought of as a potential root point related to v , but it is not if v is too close to the actual root position. The DFS-distance transformation is applied then again with s as the seed, and its opposite feature point p is obtained. p is set as the root point if its radius is bigger; or else s is.

Concurrently with the specification of the root point, all tip features are detected also based on the DFS-distance transformation to the root point. The tip feature of each branch of the volume is detected from the local maximum cluster on the root point distance map. The number of all tip points is noted as N in this subsection.

6.2 Distance transformation from multiple source points

The DFMS transformation (distance from multiple source points) proposed in (Xiang et al., 2008) takes multiple DFS transformations regarding each seed point as a tip; the distance evaluation scheme is 1-2-3.

Ramification points are calculated with the method in Subsection 4.2. In the process of DFMS transformation, a DFS-distance transformation stops once a ramification point r is met. Then, all the DFS-distance transformations at the ramification node compete against each other to propagate toward the root point. The losers stop while the winner continues until another ramification or the root point is met.

Fig. 12. Decomposition based on path competition; (a) three paths compete for growing; (b) *Path 1* and *Path 3* meet at the ramification *A*; (c) *Path 1* wins and continues to compete with *Path 2*; (d) *Path 1* wins all and become the trunk

The number of DFS-distance transformations corresponds to that of components passing the ramification point r . This number, denoted as $m(r)$, is saved as an index of r . All these DFS-distance transformations will be reused for efficient skeletonization.

Fig. 12 demonstrates the procedure of DFMS transformations for decomposition on a volume model with three branches. Fig. 12 (a) shows that three paths start from three tips, respectively. Fig. 12 (b) shows that a path stops growing when it reaches a ramification. *Path*

1 and *Path* 3 meet at the ramification *A* and only one branch is left, along which both paths compete for growing. *Path* 2 grows until meeting the ramification *B*, waiting other path to meet with. In Fig. 12 (c), *Path* 1 wins over *Path* 2 and it continues growing until reaching *B*. In Fig. 12 (d), *Path* 1 wins over *Path* 2 and it continues growing until all voxels in the volume are traversed. In fact, volume decomposition finishes when *Path* 1 meets the root.

6.3 Optimisation of segmentation surface

Segmentation at ramifications plays an important role in volumetric decomposition to obtain smooth and appropriate cutting boundaries for different components. Our segmentation method is based on the DFMS transformation above, and it is an extension of the approach in Subsection 5.1.

Fig. 13. Segmentation at a ramification; (a) direction vector; (b) Segmentation without optimisation; (c) Segmentation with optimisation

For simplicity, we only describe the cut-surface for a sub-branch corresponding to a tip point s . The ramification cluster found by a DFS-distance transformation with tip s as the seed can be divided into some self-connected groups, shown as two sets in green in Fig. 13(a). In Fig. 13 (a), the green regions stand for the neighbour clusters at the ramification, whose DFS-distance value is larger by 1 than that of the ramification cluster. We connect their centres and obtain a direction vector as the black arrow between them. Along this direction, we search for all the voxels on the shortest path linking the two sets. After cutting off these voxels, we mark the left in the clusters as parts belonging to the sub-branch indicated by the starting tip feature. This is the process of the optimisation of a cut-surface. The segmentation result around a ramification before using optimisation is demonstrated in Fig. 13(b) and the one after optimisation is in Fig. 13(c). The optimisation gives a better segmentation in Fig. 13 (c) than without the optimisation in Fig. 13 (b). The above algorithm is based on an assumption that components are approximately cylindrical.

6.4 Path growing for shape segmentation

From N tip features, we have N paths corresponding to N components, respectively. When a voxel is reached by a certain path, we mark the voxel with the feature component corresponding to this path. Those paths growing from tips to the root point will stop when a ramification appears. Growing here does not mean the development of plant architecture, but the voxel propagation from a tip point toward the root point until the specified conditions are satisfied. Rules for path growing are as follows:

- Step 1) A path grows from its tip point of each component. Growing stops when the path meets a ramification. The state of a growing is either continuing, waiting or stopping; all voxels on the path are labelled with the index of the tip point;
- Step 2) For each ramification point r , all paths reaching it are checked. Let $m=m(r)$. If at least two paths are not indexed, they are waiting here. If only one path is not, all the other $m-1$ paths will compete. The winner will continue to grow, but the others fail. The winning criterion is that the distance to a tip point is the biggest than those to other tips.
- Step 3) All paths continue to grow until a new ramification is met, and return to Step 2;
- Step 4) If only one path can grow currently, the path will expand in the direction of increasing distance. If all voxels are indexed, decomposition finishes. If there are un-indexed voxels left and no paths can grow, a loop appears in the shape. They are classified as new components according to connectivity;
- Step 5) All cutting boundaries are optimised.

The result of volume segmentation based on path growing is a hierarchical structure too. It represents the topological structure of the shape and it will be used to construct the skeleton hierarchy.

6.5 Efficient extraction of smooth skeletons

So far, the branched volume has been decomposed into N components, each of which corresponds to a tip feature. Here, we proceed to extract skeletons for each sub-branch component.

In voxel space of each component, DFS transformation with respect to its tip point is useful. Since DFS transformations have been calculated in decomposition, however, they can be reused here for efficient computation.

The distance value of each cluster will increase when path growing. The barycentre of each cluster is calculated and considered as a skeleton node since the topology of each component is simple. Each node has the distance value one larger or smaller than its neighbour nodes; hence all skeleton nodes are connected in distance-increasing order. Each skeleton is from a tip point to a ramification or the root point.

Because the connection of neighbour nodes may produce a large number of skeleton points. An appropriate cluster interval, as the distance sampling in (Xu et al., 2007), is adopted to generate equivalent skeleton point range for each sub-branch, in order for both less skeleton points and higher data compression rate. A cluster interval is a group of consecutive clusters in a component. A skeleton node is the barycentre of all the points in a cluster interval. The application of interval is a filtering of skeleton shape.

In order to deal with noisy data robustly, we compute a skeleton point $\bar{X}_t$ of a cluster interval as

$$\bar{X}_t = w_1 \left[\frac{(\bar{X}_{t-1} - \bar{X}_{t-2}) \|\bar{C}_t - \bar{X}_{t-1}\|}{\|\bar{X}_{t-1} - \bar{X}_{t-2}\|} + \bar{X}_{t-1} \right] + w_2 \bar{C}_t \quad (6)$$

where $\bar{C}_t$ is the barycentre of the cluster interval, $\bar{X}_{t-1}$ and $\bar{X}_{t-2}$ denote the skeleton nodes in the last two cluster intervals respectively, and w_1 and w_2 are two weighting coefficients.

The first term in (6) is a smoothness constraint. For each skeleton node, its radius is evaluated based upon the projections of voxels of the cluster on the plane, through the cluster centre and perpendicular to the skeleton.

For robustness to noises, we re-compute the radii along the skeleton varying too seriously. In our following experiments, w_1 and w_2 are set as 0.4 and 0.6 respectively.

Finally, skeletons of components are connected according to the hierarchical structure obtained in volume decomposition in Subsection 6.4.

6.6 Application to Tree Branch Reconstruction

To reconstruct real plants based on skeletonization, the point cloud data of two real bonsai trees are acquired by laser scanning from six angles (Fig. 19 and Fig. 20). They are then registered in a single coordinate system. The volume data of a tree is obtained from the resultant point cloud using an octree space subdivision. All skeletons of the volume data are extracted, where each skeleton node is accompanied with a radius representing its distance to the boundary. A generalized cylinder is then constructed based on each skeleton. Thus a mesh model is available when each cylinder is converted to polygons.

6.7 Experiments on efficient skeletonization

We make experiments upon the proposed algorithms of volume decomposition, efficient skeletonization and surface reconstruction using virtual volume models and those converted from laser scan data. Virtual plant models without leaves are generated by the software AMAP Gensis™. Other virtual models are generated from mathematical formula.

Fig. 14. Synthetic Sunflower branched volume; (a) the volume data; (b) volume decomposition; (c) extracted skeletons; (d) a zoom-in to (b); (e) a zoom-in to (c)

Fig. 15. Synthetic sweep willow branched volume; (a) the volume data; (b) volume decomposition; (c) extracted skeletons; (d) a zoom-in to (b); (e) a zoom-in to (c)

Fig. 14 and Fig. 15 present the results about two synthetic data sets: a virtual sunflower branched volume model and a virtual sweep willow branched volume model. Different colours are adopted to distinguish distinct components. It can be seen that the algorithms decompose the branches into components with appropriate boundaries and the extracted skeletons preserve the volume shape.

Fig. 16 shows the result of skeletons of a virtual colon shell volume data generated by a generalized cylinder spanned from an interpolation curve. The black curve is the extracted skeleton while the blue one is interpolation curve defining the volume surface. It can be seen that the extracted skeleton is very similar to the mathematical skeleton of the cylinder, and its centricity is well kept if the shape curves sharply.

Fig. 16. Synthetic volume data in two views: a virtual colon; (a) the volume and its skeleton curves; (b) a new view of (a)

Fig. 17 Synthetic volume data: a mechanical part with a loop; (a) the volume; (b) the extracted skeleton with voxel points as the background; (c) a new view of (a); (d) a new view of (b); (e) a close view of (d).

Fig. 17 shows the results of volume decomposition and skeletons of a synthetic data generated by straight lines and circles. It can be seen that the new approach works robustly on such a shape with a loop, and the volume decomposition is consistent to human perception. The extracted skeletons reflect the topology of the volume and they are smooth.

Fig. 18. Comparison on the results of skeletonization; (a) The result of (Zhang et al., 2008); (b) The result of (Xiang et al., 2008)

A comparison is made in Fig. 18 on the qualities between by volume decomposition and skeleton extraction of (Zhang et al., 2008) and by those of (Xiang et al., 2008). It displays that the skeletons in Fig. 18(a) are smoother and more centred than those in Fig. 18(b). In their zoom-in figures, the skeletons extracted by (Xiang et al., 2008) is connected more properly and better centred in the volume than those by (Zhang et al., 2008).

Fig. 19 and Fig. 20 demonstrate the experimental results of the reconstruction of real bonsai trees with high fidelity. Fig. 19 (a) and Fig. 20 (a) are the raw volume data from laser scan and those of (b) are decomposition volume data. Their (c) and (d) are extracted skeletons and final reconstructed mesh models of the branches, respectively. It can be seen that the reconstruction keeps well the tree shape noise-robustly.

Fig. 19. Reconstruction of a bonsai tree *murraya*; (a) raw data after the detection of tip features; (b) decomposition results; (c) extracted skeletons; (d) final reconstructed models;

Fig. 20. Reconstruction of a bonsai tree *asclepiadaceae*; (a) raw data after the detection of tip features; (b) decomposition results; (c) extracted skeletons; (d) final reconstructed models;

Table 2 lists the details about the data with our efficient skeletonization algorithm of the six examples, which are the number of voxels in the volume, the number of branches of the data, the skeleton sample steps, and the number of skeleton nodes.

Name of volume model	Number of Voxels	Number of Branches	Number of Clusters in an Interval	Number of Skeleton Nodes
Sunflower (Fig. 14)	12,367	13	10	203
Willow (Fig. 15)	49,883	18	10	297
Virtual Colon (Fig. 16)	15,392	1	10	40
Part with a loop (Fig. 17)	3,361	5	5	71
Tree <i>murraya</i> (Fig. 18)	3,788	12	5	181
Tree <i>asclepiadaceae</i> (Fig. 19)	2,746	13	10	99

Table 2: Volume and skeleton specifications with efficient skeletonization algorithm

Fig. 21 Volume decomposition of four synthetic tree volume when segmentation surfaces are optimized; (a) a prone willow tree; (b) a holly tree; (c) a poplar tree; (d) a will tree

Fig. 21 and Fig. 22 show the results of volume decomposition and skeleton nodes of the volume data of four virtual trees with more complex topology. We can see that our methods still work well.

Fig. 22 skeleton nodes of the four synthetic tree volumes in Fig. 17; (a) a prone willow tree; (b) a holly tree; (c) a poplar tree; (d) a will tree

7. Conclusion and Future Work

We proposed algorithms on volume decomposition and hierarchical skeletonization for branched objects in (Zhang et al., 2008). The extracted skeletons are highly centred. One important advantage of our work is the help of volume decomposition to hierarchical skeletons, so that the final extracted skeletons keep the structure of the volume data well.

This work of (Zhang et al., 2008) is extended in (Xiang et al., 2008; Ma et al., 2008; Ma et al., 2009) with the following contributions: segmentations are optimised at ramifications; extracted skeletons are smoother and more robust; the new method is applicable to shapes with a loop and shell volume models; it is more efficient since skeleton reconnection is unnecessary. The experiments are of high fidelity on reconstruction of real *bonsai* trees

We plan to continue our work in three aspects: internal navigation, segmentation of mesh models based on skeletons and the reconstruction of big trees. We would like to guide internal navigation hierarchically inside human organs with a complex topological structure (Bartz, 2005). The work of reconstructing real big trees from laser scan data in (Xu et al., 2007) and (Cheng et al., 2007) could be improved with shape decomposition. To avoid occlusion of laser scan data for the reconstruction of big real trees, the technique of particle flow can be applied. Like the work of (Reniers & Telea, 2007), this approach can also be applied to shape animation in our future work.

8. Acknowledgement

This work is supported in part by National Natural Science Foundation of China with projects No. 60672148 and No. 60872120; in part by the National High Technology Development 863 Program of China under Grant No. 2008AA01Z301, No. 2006AA01Z301, and 2008AA10Z218; in part by Science and Technology Commission of Shanghai Municipality under Grant No. 08511501002; in part by the project; and in part by the Project Arcus 2006 Languedoc-Roussillon/Chine.

9. References

- Au, O.; Tai, C.; Chu, H.; Cohen-Or, D. & Lee, T. (2008). Skeleton extraction by mesh contraction. *ACM Transactions on Graphics*, Vol.27, No 3, pp. 1-10, ISSN: 0730-0301.
- Amenta, N.; Choi, S.; Kolluri R. (2001) The power crust. in *Proceedings of the sixth ACM symposium on Solid modeling and applications*, pp. 249 - 266; ISBN:1-58113-366-9; Ann Arbor, Michigan, United States; June , 2001; ACM New York, NY, USA
- Bartz, D. (2005). Virtual Endoscopy in Research and Clinical Practice, *Computer Graphics Forum* , Vol.24, No.1, pp. 111-126, ISSN: 0167-7055.
- Bitter, I.; Sato, M.; Bender, M.; Mcdonnell, K.; Kaufman, A. & Wan, M. (2000). Ceasar, A smooth, accurate and robust centerline extraction algorithm. In *Proc. Visualization 2000*, pp. 45-52, ISBN: 0-7803-6478-3; Salt Lake City, Utah, United States; October 2000; IEEE Computer Society Press Los Alamitos, CA, USA.
- Bitter, I.; Kaufman, A. & Sato, M. (2001). Penalized-distance volumetric skeleton algorithm. *IEEE Transactions on Visualization and Computer Graphics* , Vol.7, No.3, pp. 195-206, ISSN: 1077-2626.
- Brostow, G. J. ; Essa, I. ; Steedly, D. & Kwatra, V. (2004). Novel skeletal representation for articulated creatures. In *Proceedings of the European Conference on Computer Vision (ECCV04)*, vol. III, pp.66-78, ISBN: 3-540-21984-6, Prague, Czech Republic; May, 2004; Springer-Verlag, Berlin Heidelberg.
- Cheng, C.; Zhang, X. & Chen, B. (2007). Simple reconstruction of tree branches from a single range image. *Journal of Computer Science and Technology* , Vol.22, No.6, pp. 846-858, ISSN: 1000-9000 (print version) ISSN: 1860-4749 (electronic version).
- Cornea, N.; Silver, D., Yuan, X. & Balasubrama-Nian, R. (2005). Computing hierarchical curve-skeletons of 3d objects. *The Visual Computer* , Vol.21, no.11, pp. 945-955, ISSN: 0178-2789 (print version), ISSN: 1432-2315 (electronic version).
- Cornea, D. & Min, P. (2007). Curve-skeleton properties, applications, and algorithms. *IEEE Transactions on Visualization and Computer Graphics* , Vol.13, No.3, pp. 530-548, ISSN: 1077-2626.
- Funkhouser, T.; Kazhdan, M.; Shilane, P.; Min, P., Kiefer, W.; Tal, A.; Rusinkiewicz, S. & Dobkin, D. (2004). Modeling by example, *ACM Transactions on Graphics*, Vol. 23, No. 3, 652-663, ISSN: 0730-0301.
- Gagvani, N. & Silver, D. (1999). Parameter controlled volume thinning. *Graphical Models and Image Processing* , Vol.61, no.3, pp.149-164, ISSN: 1049-9652.
- Giblin, P. & Kimia, B. (2004). A formal classification of 3d medial axis points and their local geometry *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 26, No.2, pp. 238-251, ISSN: 0162-8828.

- Ju, T.; M. Baker, L. & Chiu, W. (2007). Computing a family of skeletons of volumetric models for shape description. *Comput. Aided Des*, Vol.39, No.5, pp. 352-360, ISSN: 0010-4485.
- Katz S. & Tal A. (2003). Hierarchical mesh decomposition using fuzzy clustering and cuts. *ACM Computer Graphics (Proc. of SIGGRAPH 2003)* , Vol. 22, No. 3, 954-961, ISSN: 0730-0301.
- Li, X.; T. Woon, W.; Tan, T. S. & Huang, Z. (2001), Decomposing polygon meshes for interactive applications. In *I3D '01: Proceedings of the 2001 symposium on Interactive 3D graphics*, pp.35-42, ISBN:1-58113-292-1, Research Triangle Park, North Carolina, USA, March 2001, ACM Press, New York, NY, USA.
- Lien, J. -M. & Amato, N. M. (2004). Approximate convex decomposition of polyhedra. In *SIGGRAPH '04: ACM SIGGRAPH 2004 Posters*, pp. 2, ISBN: 1-58113-896-2, Los Angeles, California, August 2004, ACM Press, New York, NY, USA.
- Lien, J. & Amato, N. (2005). Simultaneous shape decomposition and skeletonization using approximate convex decomposition. In *Technical Report, TR05-004*, pp. 44-47, Parasol Laboratory, Department of Computer Science, Texas A M University; College Station, Texas, USA; Dec 2005.
- Lien, J. -M. & Amato, N. M. (2007). Approximate convex de-composition of polyhedra. In *Symposium on Solid and Physical Modeling*, pp.121-131, ISBN:1-59593-358-1, Cardiff, Wales, UK; Jun 2006; ACM Press, New York, NY, USA.
- Liu, J.; Zhang, X. & Blaise, F. (2004). Distance contained centerline for virtual endoscopy. In *IEEE International Symp. Biomedical Imaging Macro to Nano (ISBI)*, pp. 261-264, ISBN: 0-7803-8388-5, Arlington, VA, USA, April 2004, IEEE Press, New Jersey, USA.
- Luboz, V.; Wu, X., Krissian, K.; Westin, C. -F.; Kikinis, R.; Cotin, S. & Dawson, S. (2005). A segmentation and reconstruction technique for 3d vascular structures, In *Proceedings Medical Image Computing and Computer-Assisted Intervention - MICCAI 2005*, pp. 43-50, ISSN: 0938-7994 (Print) 1432-1084 (Online); Palm Springs, CA, USA; October 26-29, 2005; Springer-Verlag, Berlin, Heidelberg.
- Ma, C. M. & Sonka, M. (1996). A fully parallel 3D thinning algorithm and its applications. *Computer Vision and Image Understanding*, Vol.64, No.3, pp. 420-433, ISSN: 1077-3142.
- Ma, W.; Xiang, B.; Zhang, X. & Zha H. (2008), Decomposition of Branching Volume Data by Tip Detection. In *ICIP08: IEEE International Conference on Image Processing(ICIP)*. pp. 1948-1951. ISBN: 978-1-4244-1765-0; San Diego, California, U.S.A; October, 2008; IEEE Press, New Jersey, USA.
- Ma, W.; Xiang, B.; Zha, H.; Liu, J. & Zhang, X. (2009). Modeling Plants with Sensor Data, *Science in China Series F: Information Sciences*, Vol. 52, No. 3, pp. 500-510, ISSN: 1009-2757 (Print) 1862-2836 (Online).
- Reniers, D. & Telea, A. (2007), Skeleton-based Hierarchical Shape Segmentation, in: *Proceedings of IEEE International Conference on Shape Modeling and Applications, 2007. SMI '07*. pp. 179-188; ISBN: 0-7695-2815-5; Lyon, France; 13-15 June 2007, IEEE Computer Society Washington, DC, USA
- Rom, H. & Medioni, G. (1993). Hierarchical decomposition and axial shape description. *IEEE Transactions on Pattern Analysis and Machine Intelligence* , Vol.15, No.10, pp. 973-98, ISSN: 0162-8828.

- Sato, M.; Bitter, I.; Bender, M. & Kaufman, A. (2000). Teasar. Tree-structure extraction algorithm for accurate and robust skeletons. In *Proc. 8th Pacific conf. Computer Graphics and Applications*, pp. 281-289, ISBN: 0-7695-0868-5; Hong Kong, China, October 2000; IEEE Computer Society Washington, DC, USA.
- Shahrokni, A.; Zoroofi, R. & Soltanian-Zadeh, H. (2001). Fast skeletonization algorithm for 3d elongated objects, In *Proc. SPIE 4322, Vol. 4322, Medical Imaging 2001*, pp. 323-330, ISBN: 0-8194-4008-6, February 2001, San Diego, CA, USA, SPIE Press, Bellingham, USA.
- Siddiqi, K. and Kimia, B.(1995), Parts of visual form: Computational aspects, , *IEEE Transactions on Pattern Analysis and Machine Intelligence.*, vol. 17, No. 3, pp. 239-251,. ISSN: 0162-8828
- Simmons, M. & Sequin, C. H. (1998). 2D shape decomposition and the automatic generation of hierarchical representations. *International Journal of Shape Modeling*, Vol., No.4, pp. 63-78. ISSN: 0218-6543.
- Telea A.; Sminchisescu C. & Dickinson S. (2004). Optimal inference for hierarchical skeleton abstraction. In *ICPR 2004*, pp. 19-22; ISBN:1051-4651; Cambridge, UK; August, 2004; IEEE Computer Society Washington, DC, USA.
- Tung, T. & Schmitt, F. (2004). Augmented reeb graphs for content-based retrieval of 3d mesh models. In: *Proceedings of Shape Modeling Applications 2004.*, pp.157-166, ISBN: 0-7695-2075-8; Genoa, Italy; June 2004; IEEE Computer Society, Los Alamitos, California, USA.
- Wan, M.; Liang, Z.; Ke, Q.; Hong, L.; Bitter, I. & Kaufman, A. (2002). Automatic centerline extraction for virtual colonoscopy. *IEEE Transactions on Medical Imaging* , Vol.21, No 12, pp. 1450-1460. ISSN: 0278-0062.
- Wang, T. & Basu, A. (2007). A note on 'A fully parallel 3D thinning algorithm and its applications'. *Pattern Recognition Letters*, Vol. 28, No. 4. pp. 501-506. ISSN:0167-8655.
- Xiang, B.; Zhang, X.; Ma, W. and Zha, H (2008). Skeletonization of Branched Volume by Shape Decomposition. In *CCPR08: Chinese Conference on Pattern Recognition*. pp. 116-121. ISBN: 978-1-4244-2316-3; Beijing, China; October 2008; IEEE Press, New Jersey, USA.
- Xu, H.; Gossett, N. & Chen, B. (2007). Knowledge and heuristic-based modeling of laser-scanned trees. *ACM Transactions on Graphics* , Vol. 26, No.4, pp. 19. ISSN: 0730-0301.
- Zhang, X.; Liu, J.; Li, Z. & Jaeger, M (2008). Volume decomposition and hierarchical skeletonization. In *VRCAI '08: Proceedings of The 7th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry*. ISBN:978-1-60558-335-8; Singapore; December 2008; ACM Press, New York, NY, USA.
- Zhang, X.; Liu, J.; Jaeger, M. & Li, Z. (2009). Volume Decomposition for Hierarchical Skeletonization, *The International Journal of Virtual Reality*; to appear
- Zhou, Y. & Toga, A. W. (1999). Efficient skeletonization of volumetric objects. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 5, No.3, pp. 196-209. ISSN: 1077-2626.

IntechOpen

IntechOpen

Pattern Recognition

Edited by Peng-Yeng Yin

ISBN 978-953-307-014-8

Hard cover, 568 pages

Publisher InTech

Published online 01, October, 2009

Published in print edition October, 2009

For more than 40 years, pattern recognition approaches are continually improving and have been used in an increasing number of areas with great success. This book discloses recent advances and new ideas in approaches and applications for pattern recognition. The 30 chapters selected in this book cover the major topics in pattern recognition. These chapters propose state-of-the-art approaches and cutting-edge research results. I could not thank enough to the contributions of the authors. This book would not have been possible without their support.

How to reference

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Xiaopeng Zhang, Bo Xiang, Wujun Che and Marc Jaeger (2009). Volume Decomposition and Hierarchical Skeletonization for Shape Analysis, Pattern Recognition, Peng-Yeng Yin (Ed.), ISBN: 978-953-307-014-8, InTech, Available from: <http://www.intechopen.com/books/pattern-recognition/volume-decomposition-and-hierarchical-skeletonization-for-shape-analysis>

INTECH
open science | open minds

InTech Europe

University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China

Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

© 2009 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the [Creative Commons Attribution-NonCommercial-ShareAlike-3.0 License](https://creativecommons.org/licenses/by-nc-sa/3.0/), which permits use, distribution and reproduction for non-commercial purposes, provided the original is properly cited and derivative works building on this content are distributed under the same license.

IntechOpen

IntechOpen