

RESEARCH ARTICLE

Analytical solutions for tree-like structure modelling using subdivision surfaces

Xiaoqiang Zhu^{1,2}, Xiaogang Jin^{1*} and Lihua You³¹ State Key Lab of CAD&CG, Zhejiang University, Hangzhou 310058, China² School of Communication & Information Engineering, Shanghai University, Shanghai 200444, China³ National Center for Computer Animation, Bournemouth University, Poole, BH12 5BB, UK

ABSTRACT

We present a novel approach to efficiently modelling branch structures with high-quality meshes. Our approach has the following advantages. First, the limit surface can fit the target skeleton models as tightly as possible by reversely calculating the control vertices of subdivision surfaces. Second, high performance is achieved through our proposed analytical solutions and the parallel subdivision scheme on a graphics processing unit. Third, a smooth manifold quad-only mesh is produced from the adopted Catmull–Clark scheme. A number of examples are given to demonstrate applications of our approach in various branch structures, such as tree branches, animal torsos, and vasculatures. Copyright © 2013 John Wiley & Sons, Ltd.

KEYWORDS

branching structure modelling; Catmull–Clark subdivision; limit surface; CUDA

*Correspondence

Xiaogang Jin, State Key Lab of CAD&CG, Zhejiang University, Hangzhou, China.

E-mail: jin@cad.zju.edu.cn

1. INTRODUCTION

Branch structures can be used to represent ubiquitous three-dimensional (3D) models, such as tree trunks, vessel systems, animal torsos, and so on. Shapes with branch structures have natural line skeletons, which are usually taken as excellent abstracts of models. Various approaches have been developed to produce skeletons with radius information, such as interactive techniques for small models [1], contracting meshes with Laplacian operator to extract skeleton for large models [2,3], and the vessel-tracking algorithm for medical computed tomography data [4]. Efficient reconstruction of 3D models from extracted skeleton with high-quality surfaces is important for the objects with a branch structure. It has a wide range of applications in medical education, therapy planning, video games, and other purposes.

Among various existing approaches of surface reconstruction from skeletal data, the convolution surface-based approach [5–7] generates a smooth surface but involves huge potential field computation and iso-surface extraction. In addition, it usually generates low-quality triangle meshes. Although the cylinder-based representation [8] has high efficiency, it produces cracks between adjacent components. Another more advanced explicit

reconstruction technique is the Catmull–Clark subdivision, which can achieve a smooth surface and efficient computation through the exploration of its natural parallelism. The technique of reconstructing branch structures using the Catmull–Clark subdivision is proposed in [9]; however, the shrinkage problem during subdivision is not discussed. Although the approximation of implicit surface defined by metaballs is introduced by Ji *et al.* [1] to avoid shrinkage, the huge computation hinders its interactive use, particularly for large models.

Inspired by these works, we extend and improve the scheme of B-mesh [1] and make it more suitable for reconstruction of branch structures defined by the given centrelines and radii. A high efficiency can be easily achieved by parallelling the time-consuming subdivision on a graphics processing unit (GPU) and through avoiding the most computation-intensive implicit approximation. Because of our closed-form solutions for reverse calculation of the initial control mesh, the shrinkage problem is solved. In summary, our paper has the following contributions:

- (1) A parallel scheme for skeleton-based reconstruction of tree-like shapes using subdivision surfaces, which produces compact quad-only meshes

- (2) Closed-form solutions for reverse calculation of the control mesh for the Catmull–Clark subdivision

2. RELATED WORK

Our proposed approach is related to tree structures and subdivision surfaces. In this section, we briefly review existing research activities in these two fields.

2.1. Tree Structures

Recently, reconstruction of trees from laser point clouds was developed. In [10], the line skeletons and their radii are extracted from given point sets, and then a ring is placed around each node for producing a triangle mesh by connecting these rings. In [11–13], the models of the trunks and the main branches are represented with cylinders or independent mesh components. Lluch *et al.* [14] gave an approach of a single-mesh creation by refining branching areas.

There are quite a few organ shapes with tree-like structures, including vasculatures, bronchial trees, and so on. Haln *et al.* [8] utilised blended concatenated truncated cones for smoothness at branching points. Ritter *et al.* [15] explored hatching lines and silhouettes for enhancing shape perception. Convolution surfaces are a preferred option for generating smoother surfaces. Oeltze and Preim [5,6] presented a method for visualisation of anatomic tree-like structures that represents the structure models with convolution surfaces. Pihuit *et al.* [16] also made use of convolution surfaces for sketch-based modelling of vascular systems. The work is a first step towards interactive teaching of anatomy. Another approach for producing smooth surfaces of vasculature is subdivision surfaces [9,17], which is considered as one of the most advanced explicit reconstruction technique [18]. B-mesh [1] and skeleton to quad-dominant polygonal mesh [19] run on CPU are proposed to convert skeletal structures to quad-only and quad-dominant meshes, respectively.

2.2. Subdivision Surfaces

As important modelling tools, the Catmull–Clark [20], Loop [21], and Doo–Sabin [22] subdivisions have been included in such packages as CGAL [23] and OpenMesh [24] and are all performed on CPU. Recently, the massive computational capability of GPUs and high parallelism of subdivisions have given rise to many parallel subdivision algorithms [25,26]. One kind of parallel scheme is designed specially for Compute Unified Device Architecture (CUDA) [27,28]. The paper borrows the proposed parallel Catmull–Clark subdivision in [27].

Another important aspect about subdivision is to compute the limit positions of the control mesh [29,30]. To approximate the target shape with subdivision surfaces, some researches are carried out by reversing subdivision

rules [31–36]. Inspired by these studies, we give a closed-form solution for approximating tree-like structures with subdivision surfaces.

3. OVERVIEW OF APPROACH

The major task of our work is to present a scheme of skeleton-based subdivision surface modelling for tree-like models. The developed system consists of the following stages:

- *Input skeletons.* Our system begins with connected graph-based skeletons with radius information. Graph-based skeletons are natural abstracts of 3D shapes. The embedded skeletal information can be obtained from different sets of collected raw data. Many approaches can be utilised to create such skeletons conveniently. For example, skeletons of small models can be obtained through the modelling packages [1]. Meshes are one of the most preferred models in the field of computer graphics because of the hardware support for fast rendering. It is efficient to extract skeleton from large models by contracting the meshes with a Laplacian operator [2,3] (Figure 1(a–c)). Such skeletonisation can automatically create a graph-based skeleton and a by-product of mapping vertices (Figure 1(b)), from which the radii of skeletal nodes can be calculated by averaging the distances between each node and its mapping vertices.
- *Control mesh generation.* In this step, the methods in [1,9] can be used to create a control mesh based on the extracted skeletons, and the technique in [1] is utilised in our system because of its simplicity. There may be a few triangles (Figure 1(d)) at branches in the produced quad-dominant control mesh. The proposed reverse control mesh calculation makes sure that the subdivided mesh visually approximates the original mesh.
- *Subdivision.* A parallel Catmull–Clark subdivision of the control mesh is performed with CUDA on GPU, which generates the refined quad-only mesh (Figure 1(e–f)) with high efficiency.

4. CONTROL MESH GENERATION

In this section, we describe in detail the control mesh generation based on the line skeletons, especially the closed-form solutions for the reverse calculation of the control vertices.

4.1. Topology Generation

According to the description in [1], the topology of the control mesh can be divided into the following three steps:

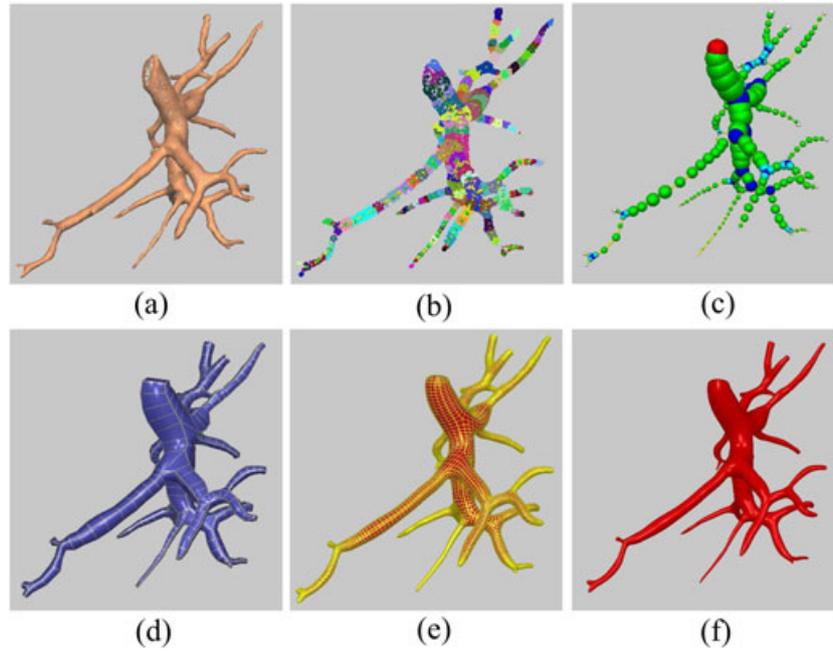


Figure 1. The overview of our approach. Given an original vasculature mesh (a), the by-product of mapping information (b) can be produced, from which embedded line skeletons with adaptive radii (c) are extracted, and the control mesh (d) is produced on the basis of the skeleton information, and then two levels of subdivision are performed (e and f).

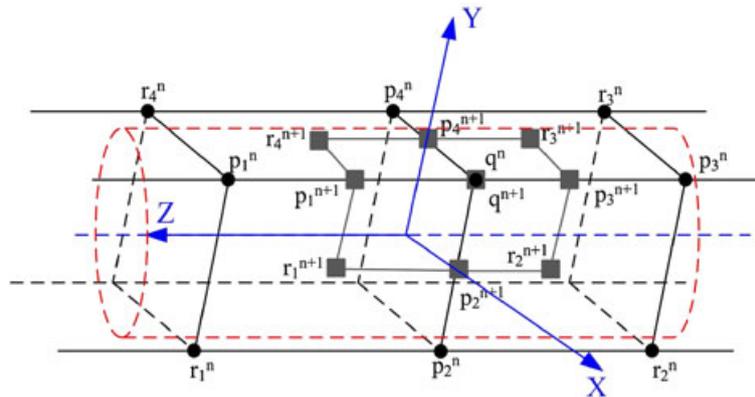


Figure 2. One step of subdivision performed on the vertex q^n and its one-ring neighbourhood. The subscript index indicates the vertex number, and the superscript is the level of subdivision.

Sweeping. After creating a cross section of a square at the root node, sweeping of the cross section along the skeleton is performed except for branching nodes.

Convex hull creation. A convex hull is created around each branching node with the cross section vertices of its parent and its children. Then, two adjacent triangles are merged into a quadrangle if the difference of their normals is small enough.

Stitching. Finally, the swept tubes along non-branching nodes and the convex hulls at branching nodes are stitched if they are coadjacent.

4.2. In-between Node Approximation

As illustrated in Figure 2, our derivation is based on the premise that a cylinder needs to be approximated by subdividing a quadrangular prism, and it is reasonable because the variation between neighbouring segments of biological models is not large. Therefore, the key problem is to make the vertex q^0 of the control mesh fall onto the target cylinder after subdivision. To further simplify the problem, we only consider the x -coordinate and the y -coordinate of q^0 because the z -coordinate only affects the deviation

along the z -axis direction and has no influences on the approximation precision.

Here, we set the x -axis and the y -axis parallel with the two neighbouring edges of the cross square and the z -axis in the skeletal direction. Therefore, for a cylinder segment with radius $R_{lmt} = \sqrt{2}\gamma$, only the parameter α has to be reversely calculated for determining the two-dimensional coordinates of $\mathbf{q}^0(\alpha, \alpha)$, which ultimately converges to $\mathbf{q}^\infty(\gamma, \gamma)$.

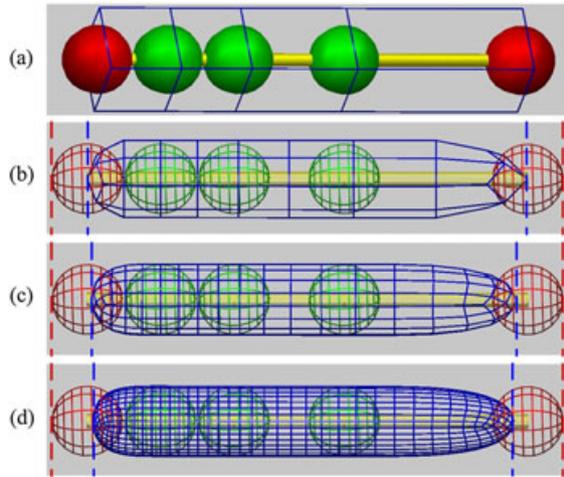


Figure 3. A cylindric skeleton with uniform radii and its subdivision approximation (green balls are in-between nodes, and red balls are end nodes). (a) The skeletal radius information and the created control mesh, (b) the first subdivision, (c) the second subdivision, and (d) the third subdivision.

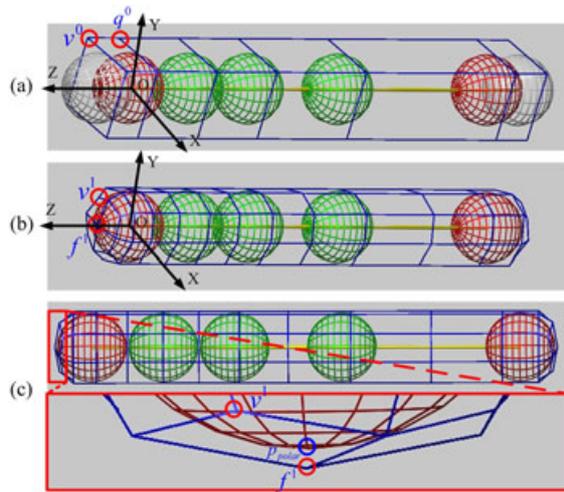


Figure 4. Reverse control vertex calculation of target cylinder (green balls are in-between nodes, red balls are end nodes, and grey balls are auxiliary nodes). (a) The created control mesh based on auxiliary nodes (grey balls), (b) the first subdivision from a perspective view, and (c) the first subdivision from an orthogonal view.

From the conclusions drawn in [29],

$$\mathbf{q}^\infty = \beta_\infty \mathbf{q}^n + \gamma_\infty \mathbf{P}^n + (1 - \beta_\infty - \gamma_\infty) \mathbf{R}^n \quad (1)$$

$$\mathbf{q}^{n+1} = a_0 \mathbf{q}^n + a_s \mathbf{P}^n + (1 - a_0 - a_s) \mathbf{R}^n \quad (2)$$

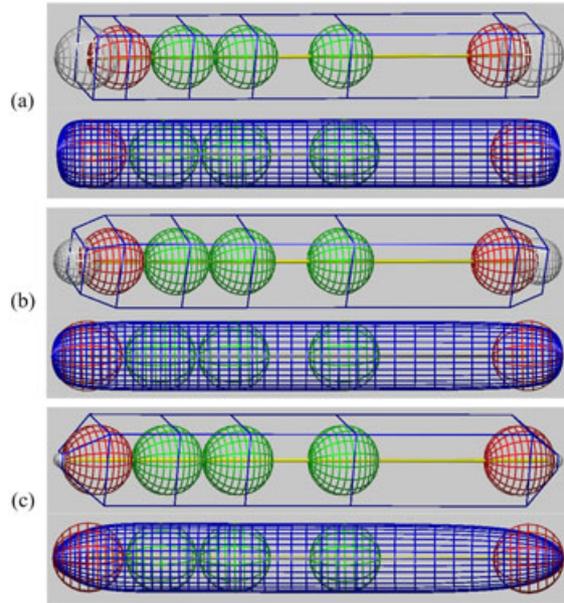


Figure 5. A three-level subdivision with t_3 to approximate the end ball in the skeletal direction and different diameter variations are achieved by adjusting the radii (equivalently s_m) of auxiliary balls. (a) $R_{\text{offset}} = R_{lmt}$, (b) $R_{\text{offset}} = 0.7R_{lmt}$, and (c) $R_{\text{offset}} = 0.2R_{lmt}$.

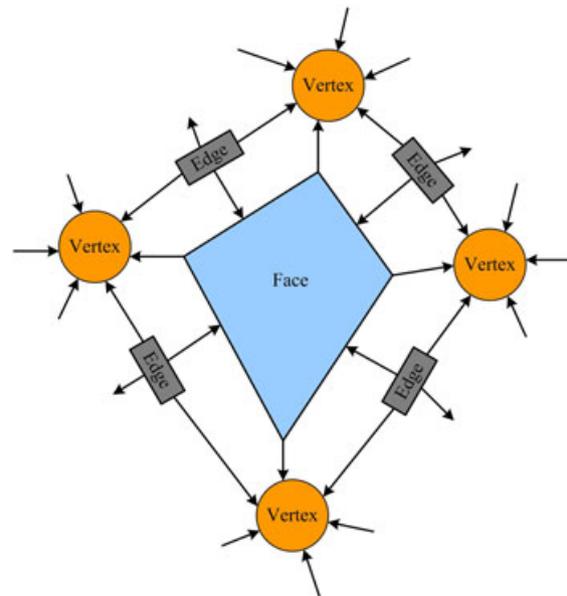


Figure 6. Topological information in the mesh structure.

$\mathbf{q}^{n+m} = \mu(m)\mathbf{q}^n + v(m)\mathbf{q}^{n+1} + (1 - \mu(m) - v(m))\mathbf{q}^\infty$
 where $\mathbf{P}^n = \frac{1}{v} \sum_{j=1}^v \mathbf{p}_j^n$ and $\mathbf{R}^n = \frac{1}{v} \sum_{j=1}^v \mathbf{r}_j^n$. For a vertex with valency of $v = 4$ in the standard Catmull–

Clark scheme, $a_0 = 1 - \frac{7}{4v} = \frac{9}{16}$, $a_s = \frac{6}{4v} = \frac{6}{16}$,
 $\beta_\infty = \gamma_\infty = \frac{4}{9}$, $\mu(m) = \frac{-1}{3}(\frac{1}{4})^m + \frac{4}{3}(\frac{1}{16})^m$, and
 $v(m) = \frac{16}{3}((\frac{1}{4})^m - (\frac{1}{16})^m)$.

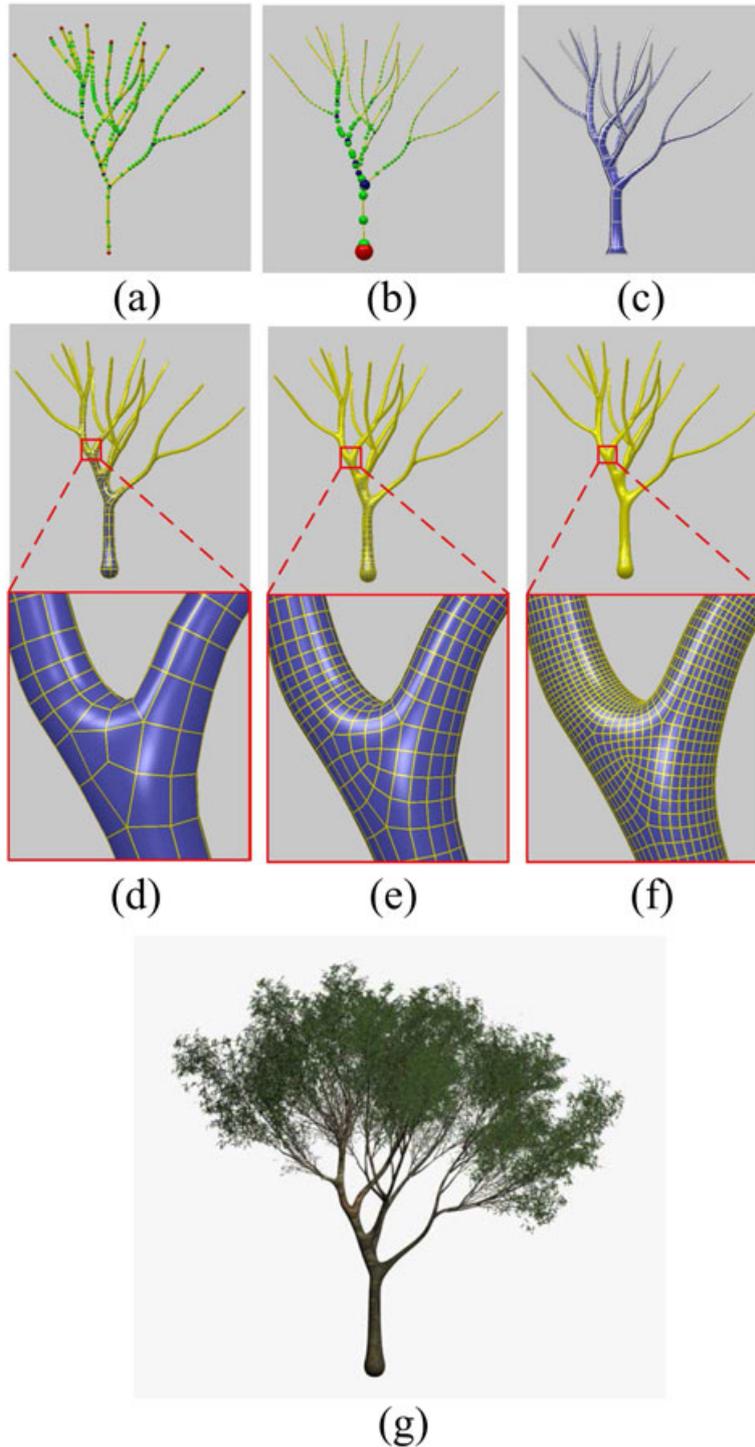


Figure 7. A tree example. (a) The extracted skeletons with uniform radius, (b) the skeleton with adaptive radius, (c) the created control mesh, (d) the first subdivision, (e) the second subdivision, (f) the third subdivision, and (g) the decorated tree with textures, tiny twigs, and leaves.

Technically speaking, it is adequate to obtain the relationship between α and γ according to Equation (1): $\alpha_\infty = \frac{3}{2}\gamma$. In our implementation, two or three levels of subdivisions have created visually smooth surfaces. Therefore, in order to make the vertices approximate the limit surface tightly after several subdivisions, we also give analytical solutions for $\alpha_m (m = 1, 2, 3, \dots)$ based on Equations (1)–(3):

$$\alpha_m = \frac{12\gamma}{8 + 4\mu(m) + \nu(m)} \quad (4)$$

where m indicates the final level of subdivision and $\alpha_1 = \frac{4}{3}\gamma$, $\alpha_2 = \frac{16}{11}\gamma$, and $\alpha_3 = \frac{64}{43}\gamma$ are mostly used in our applications. The analytical solutions of further calculations at higher levels of subdivision can be similarly obtained when required. Figure 3 illustrates a cylinder approximated by three levels of subdivision. It can be seen that the in-between green nodes are approximated very well.

4.3. End-node Approximation

Compared with the good approximation at in-between nodes, the approximation at the end nodes is very poor as illustrated in Figure 3 (red nodes). Too much shrinkage occurs in the skeletal direction, and the longer the end

skeletal segment is, the more shrinkage will be produced. In what follows, an error control scheme is presented to address this issue.

As illustrated in Figure 4(a), an auxiliary node (in grey) is appended at each end segment. Two approximation goals will be achieved. One is to guarantee the limit position of the control vertex \mathbf{v}^0 (Figure 4(a)) or the limit position of the first subdivided vertex \mathbf{v}^1 (Figures 4(b, c)) on the red ball of the end node. The other is to make the limit position of the first subdivided face point \mathbf{f}^1 (Figures 4(b, c)) approximate the polar position $\mathbf{p}_{\text{polar}}$ (Figure 4(c)) of the red ball at the end node.

A local coordinate system is created. Its origin is at the left end node (Figure 4(b)), the z -axis is in the skeletal direction, and the x -axis and the y -axis are parallel to the two neighbouring edges of the cross square, which coincide with the setting in the aforementioned section of the in-between node approximation. For the control vertex $\mathbf{v}^0(-s, s, t)$ in Figure 4(a), s and t are two parameters to be solved. The vertex position \mathbf{v}^m and the face point \mathbf{f}^m after $m (m = 1, 2, 3, \dots, \infty)$ times of subdivision can be directly computed according to Equations (1)–(3):

$$\begin{cases} \mathbf{v}^m = g_m(s, t, \alpha) \\ \mathbf{f}^m = h_m(s, t, \alpha) \end{cases} \quad (5)$$

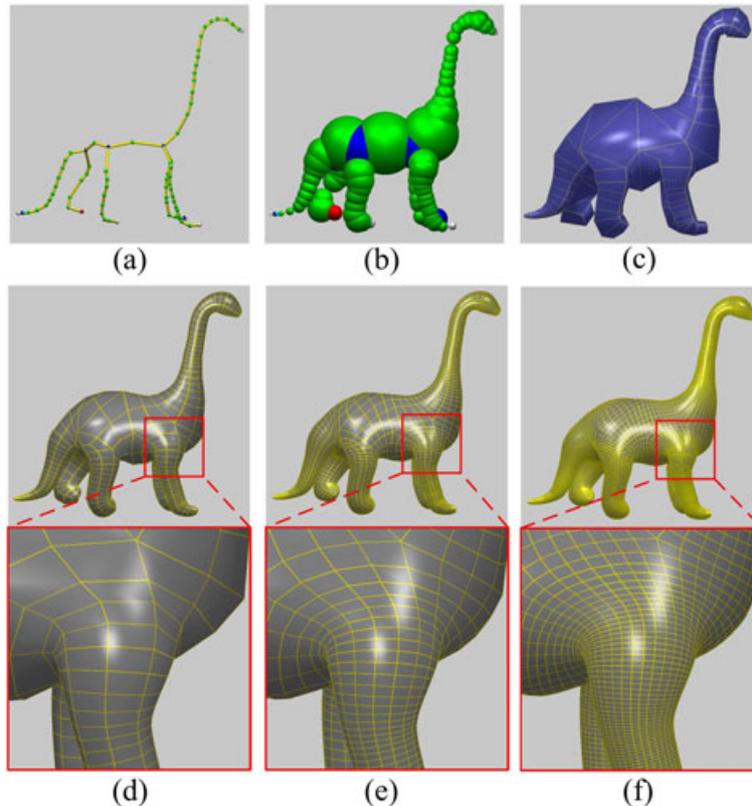


Figure 8. A dinosaur example. (a) The extracted skeletons with uniform radius, (b) the skeleton with adaptive radius, (c) the created control mesh, (d) the first subdivision, (e) the second subdivision, and (f) the third subdivision.

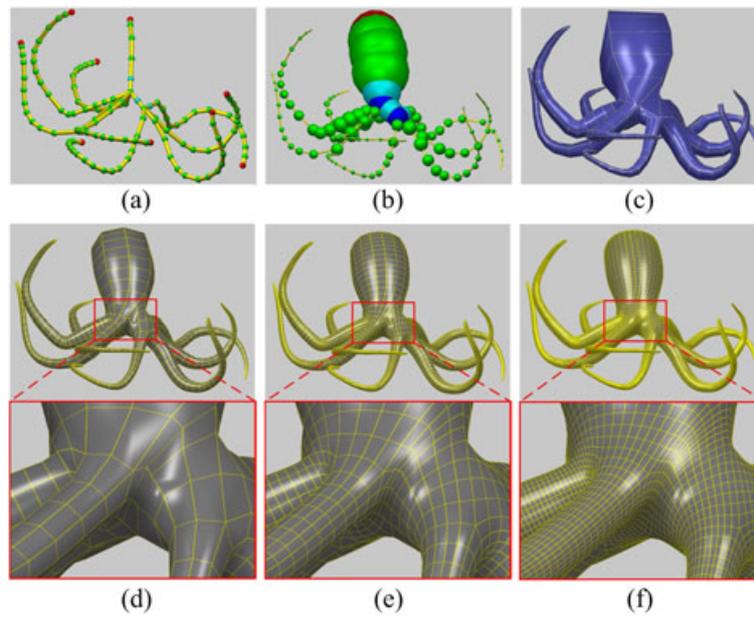


Figure 9. An octopus example. (a) The extracted skeletons with uniform radius, (b) the skeleton with adaptive radius, (c) the created control mesh, (d) the first subdivision, (e) the second subdivision, and (f) the third subdivision.

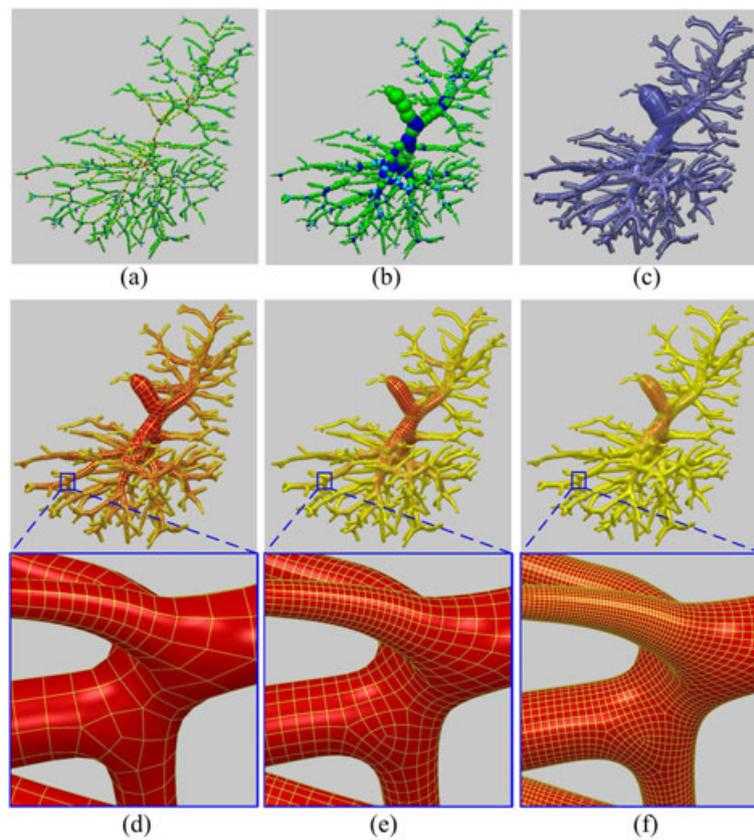


Figure 10. A vasculature example. (a) The extracted skeletons with uniform radius, (b) the skeleton with adaptive radius, (c) the created control mesh, (d) the first subdivision, (e) the second subdivision, and (f) the third subdivision.

where $\mathbf{q}^0(-\alpha, \alpha)$ is the control vertex at the end node of the line skeleton as shown in Figure 4(a), which can be calculated from Equation (5). To make the \mathbf{v}^m and \mathbf{f}^m be on the ball of the end node, we solve

$$\begin{cases} |\mathbf{v}^m| = \sqrt{2}\gamma \\ |\mathbf{f}^m| = \sqrt{2}\gamma \end{cases} \quad (6)$$

The analytical solutions for $t_m (m = 1, 2, 3, \dots, \infty)$ are given as

$$t_m = \frac{5184\sqrt{2}\gamma}{32(149 + 13\mu(m) + 101\nu(m))} \quad (7)$$

which controls the approximation precision in the skeletal direction (Figure 5). In our experiments,

$$\begin{cases} t_1 = \sqrt{2}\gamma \\ t_2 = \frac{576\sqrt{2}}{541}\gamma \\ t_3 = \frac{9216\sqrt{2}}{8521}\gamma \\ t_\infty = \frac{162\sqrt{2}}{149}\gamma \end{cases} \quad (8)$$

are the mostly used formulae in practice. The closed-form solutions for $s_m (m = 1, 2, 3, \dots, \infty)$ can be obtained as well. For the sake of conciseness, they are not presented here. As shown in Figure 5, different shapes of the subdivision surface at ends can be achieved by tuning s_m .

5. PARALLEL SUBDIVISION

Smooth surfaces can be produced by subdividing the created control mesh. Here, the parallel Catmull–Clark subdivision with CUDA is applied [27]. Because no adaptive constraints are necessary, a simplified version is enough for our application. As illustrated in Figure 1(e), excellent meshes with adaptive edge lengths to the skeletal diameters have been achieved with the standard Catmull–Clark subdivision.

The same topology structure as [27] is utilised (Figure 6). There are three main types of kernels in total in the subdivision. In the first type of *GenFacePoints* kernel, one thread is dispatched for each polygon to generate

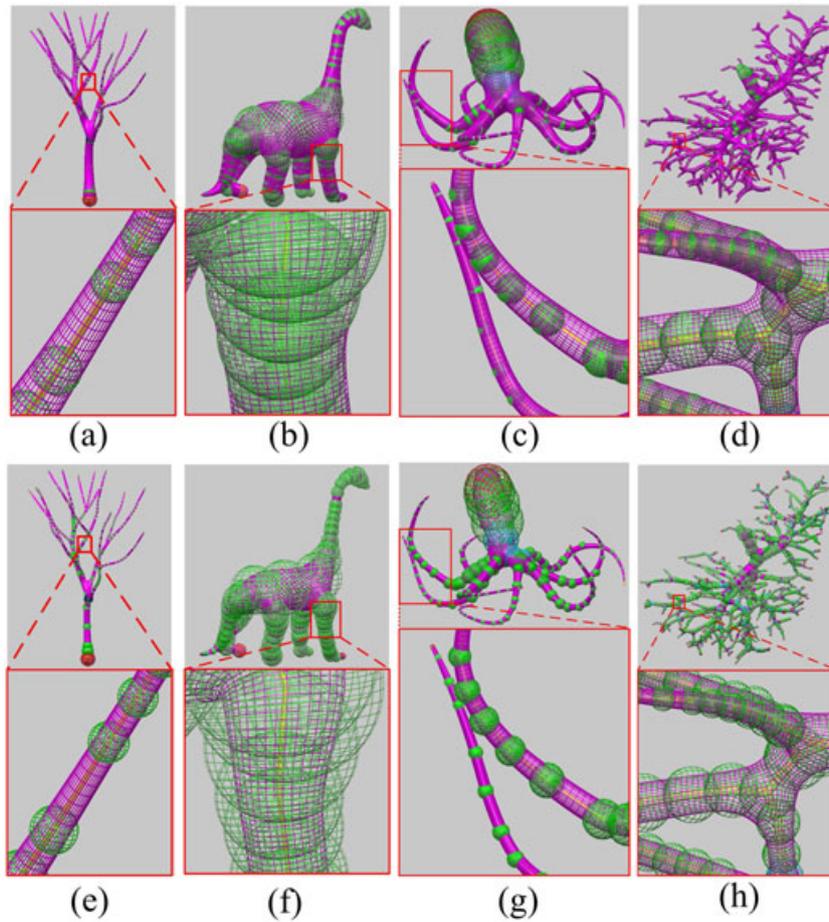


Figure 11. The advantage of our reverse control vertex calculation at in-between nodes. The green balls are in-between nodes, and the blue mesh is the three-level subdivision mesh. Our scheme approximates the target balls more accurately (a–d) than the standard Catmull–Clark subdivision (e–h), which leads to shrinkage.

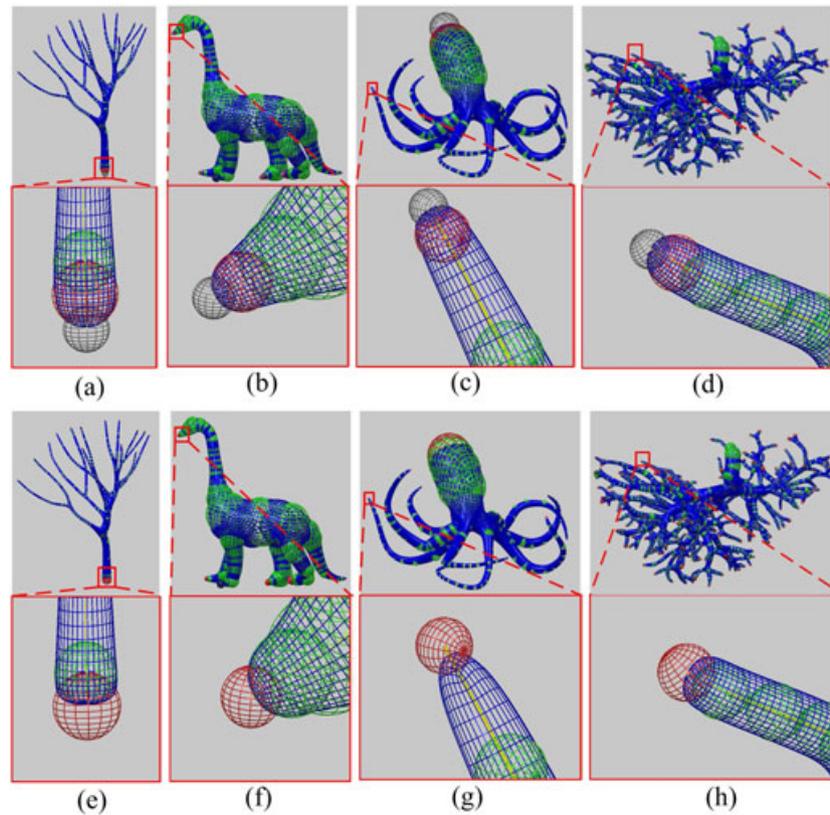


Figure 12. The advantage of our reverse control vertex calculation at end nodes. The balls in green, red, and grey are in-between nodes, end nodes, and auxiliary nodes, respectively, and the blue mesh is the three-level subdivision mesh. Our scheme with auxiliary nodes approximates the target balls in red more accurately (a–d) than that without auxiliary nodes (e–h).

a face point by averaging the vertices of the current polygon. In the second type of *GenEdgePoints* kernel, each thread is responsible for creating an edge point by averaging four vertices: the two vertices of the edge in question and the two face points of the adjoining faces. The last type of *GenVertexPoints* kernel distributes a separate thread to update each original vertex with the valence of n [20]:

$$\frac{Q}{n} + \frac{2R}{n} + \frac{S(n-3)}{n} \quad (9)$$

where Q is the average of all the face points adjacent to the old vertex, R is the average of all the mid-edge-points incident on the old vertex, and S represents the old vertex.

6. APPLICATIONS AND DISCUSSIONS

There are many tree structures in biological models. Therefore, it is important to model their shapes with excellent representation. In this section, we will give some examples to demonstrate the applications of our method. All of them are performed on a PC with an Intel Q9400 CPU (Intel Corporation, Santa Clara, CA) with 4-GB memory and

an NVIDIA GeForce GTX 260 GPU (Nvidia Corporation, Santa Clara, CA).

6.1. Plant Branch

Trees are one of the most utilised plants for realistic modelling and rendering. As a major component, trunks and branches tend to attract viewers' focuses because they shape the outline of a tree. So an efficient scheme for representing tree branches is essential especially in a large scene simulation. With our approach, a small number of quadrilaterals with nice edge flow could represent a complex tree. Such an example of eucalyptus is illustrated in Figure 7.

6.2. Animal Torso

Besides the plant trunks, some animal torsos also have a branch structure with circular cross sections. Even though highly detailed shapes are preferred in movies and video games, they are usually modelled with a coarse-to-fine procedure. That is to say, many detailed shapes are created by sculpting a base mesh that represents the rough structure of a target shape. In addition, quad meshes are often

preferred over triangle meshes. Because of the two aforementioned reasons, our method can be used to create initial quad-only meshes for adding details using the sculpting packages (Mudbox and ZBrush). Details can be seen in Figures 8 and 9.

6.3. Vasculature

Traditional iso-surface extraction from medical volume data generates triangle meshes. With such extraction, triangles with low quality are hard to avoid especially for Marching cubes. Therefore, a post-processing

step of mesh optimisation has to be used to improve the triangles. Vascular structures such as vessels and bronchial systems have natural articulated line skeletons. Our proposed scheme is especially suitable for creating 3D models of such structures as it is easy to produce quad meshes with excellent edge flows along imbedded skeletons and intuitive to manipulate the surface by editing one-dimensional skeletons. The created models can be widely applied in educational systems and therapy planning. Besides the example given in Figure 1, we here give one more vasculature example illustrated in Figure 10.

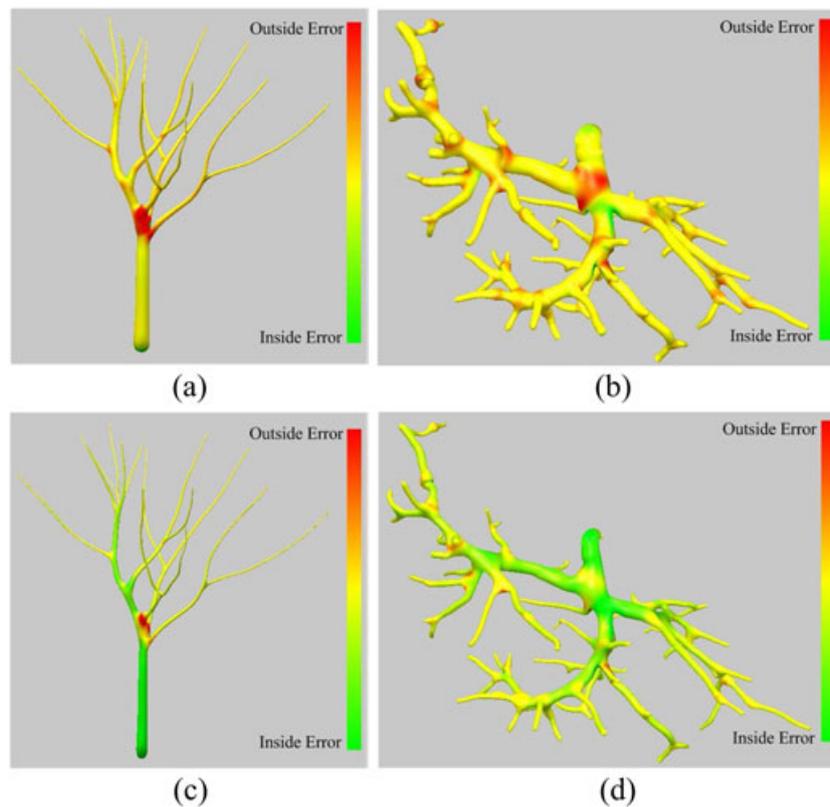


Figure 13. Approximation errors with (a and b) and without (c and d) our reverse control vertex calculation.

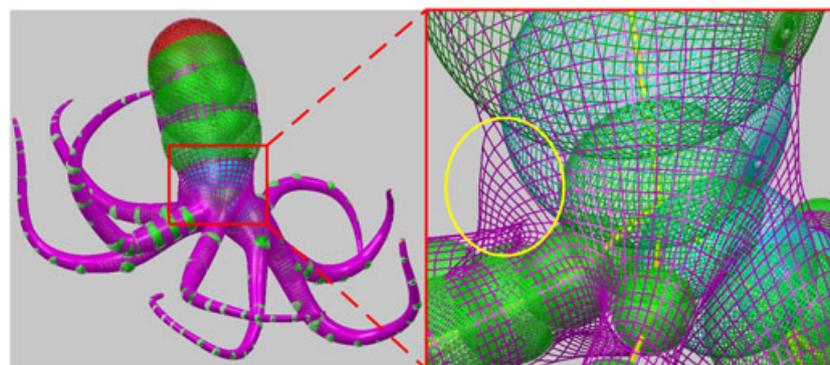


Figure 14. Large error occurs when too many branches exist.

6.4. Discussion

Compared with the standard Catmull–Clark subdivision, which leads to shrinkage (Figures 11(e–h), 12(e–h), and 15(e–h)), our scheme with reverse control vertex calculation solves the problem and approximates the target skeletal balls more accurately (Figures 11(a–d), 12(a–d), and 15(a–d)). As shown in Figure 13, our approximation effectively reduces the shrinkage problem at in-between nodes, although there are still errors at branching areas.

Our method has a drawback. The balls at branching nodes (especially the nodes with too many child branches) are usually not approximated as well as the in-between balls (Figure 14). Even though the evolution of the surface to an iso-surface defined by metaballs can obtain better approximation at both in-between nodes and branching nodes [1], the time-intensive potential field computations severely reduce the performance. Therefore, an offline distance field-based post-evolution can be employed to enhance our approach. In theory, metaballs are not suitable for producing a standard cylinder model, and

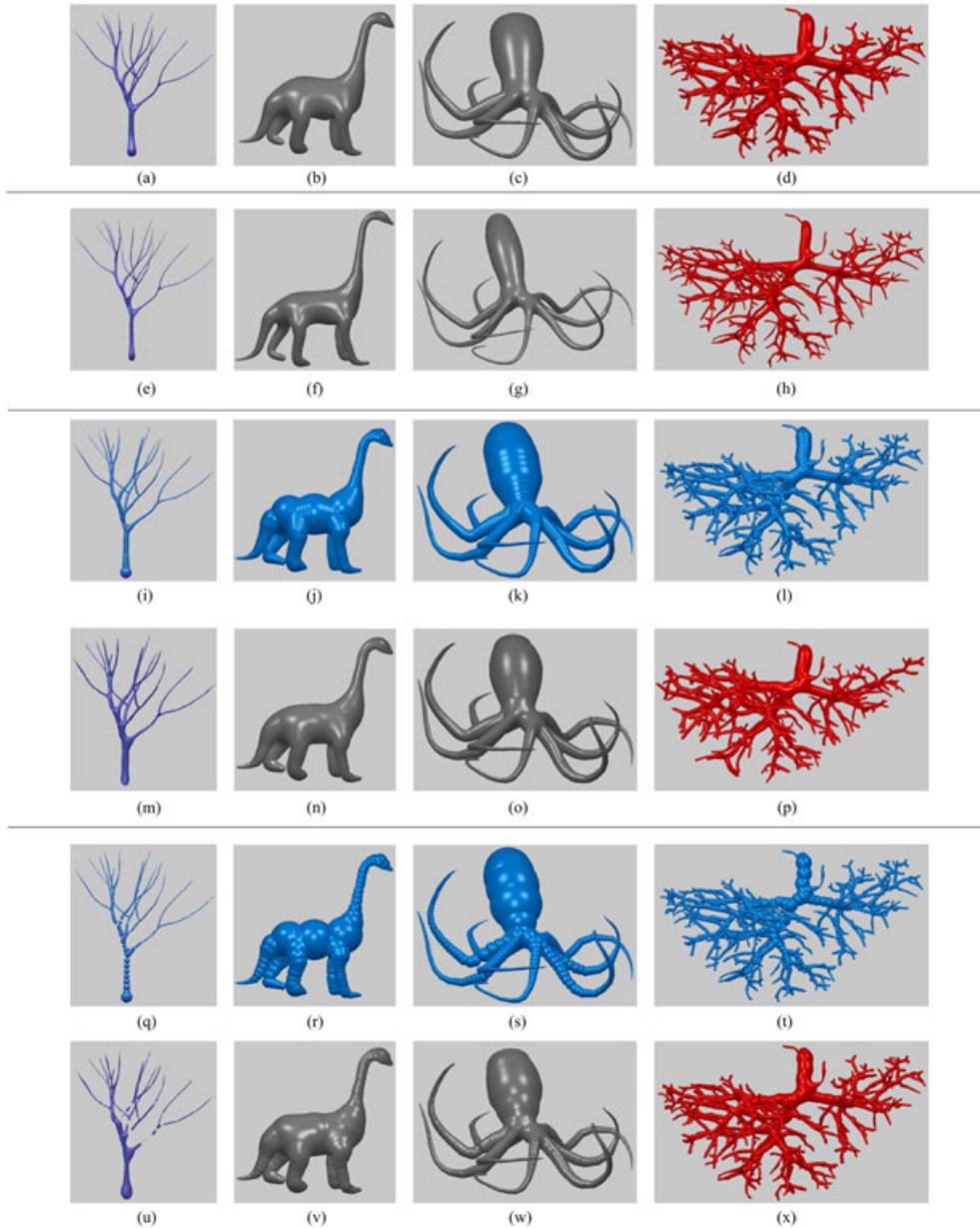


Figure 15. Our method creates nice visual effects (a–d), whereas the standard subdivision leads to shrinkage (e–h). A post-process of approximation with enough metaballs (i–l) can create smooth enough surfaces (m–p), but bulging effects occur (u–x) if insufficient metaballs are provided (q–t).

Table I. Results of the produced shapes with three levels of subdivision.

Models	Methods	Figure 15	Metaballs	Skels	Facets		Time(ms)			Total
					CtrMesh	SubMesh	CPU		Appr	
							CtrMesh	SubMesh		
Tree	Ours	(a)	—							31.8
	CC	(e)	—	253	924	58 944	10.9	20.9	—	1 209.3
	B-mesh	(m)	1 668						1 001.5	511.2
Dinosaur	Ours	(b)	—							21.4
	CC	(f)	—	84	320	20 416	5.3	16.1	—	153.5
	B-mesh	(n)	409						132.1	64.9
Octopus	Ours	(c)	—							26.1
	CC	(g)	—	132	523	33 440	8.3	17.8	—	988.7
	B-mesh	(o)	2 350						962.6	253.3
Vessel	Ours	(d)	—							183.2
	CC	(h)	—	2 145	9 110	579 200	80.9	102.3	—	30 508.3
	B-mesh	(p)	7 142						30 508.3	15 505.7
		(x)	2 615						15 322.5	

Skels, skeleton segments; CtrMesh, control mesh; SubMesh, subdivision surface mesh; Appr, approximation; CC, standard Catmull–Clark subdivision without reverse control vertex calculation; GPU, graphics processing unit.

auxiliary metaballs have to be inserted between each pair of nodes to produce a smooth field. Too many auxiliary balls (Figure 15(i–l)) require a surprising amount of calculation (Figure 15(m–p) and Table I), and insufficient auxiliary metaballs (Figure 15(q–t)) generate bulging surfaces (Figure 15(u–x)). In addition, too few metaballs will give rise to fractures (Figure 15(u)). Therefore, it is nontrivial to make a compromise between high performance and good approximation.

Besides the excellent visual effects generated with our approach, it also achieves an interactive rate easily. As shown in Table I, the whole process can be efficiently performed through parallelising the time-consuming subdivision with CUDA and preventing the most computation-intensive approximation.

7. CONCLUSIONS

In the paper, we have presented a scheme for creating the quadrilateral meshes of tree-like structures using Catmull–Clark subdivision surfaces. In order to approximate the target graph-based skeletons well, we deduced the closed-form solutions for the control vertex calculation with the reverse subdivision rules. In order to achieve high efficiency, the most time-consuming subdivision is performed in parallel with CUDA on GPU. Experiments show that our method approximates the cylinder-like shapes very well at in-between nodes and achieves visually plausible effects at end nodes. We have applied our approach to different types of models, such as trees, animal torsos, and vasculature structures, and produce excellent results for all of them.

Because the standard Catmull–Clark subdivision cannot fit a cylinder or a ball very accurately, our scheme is an approximate solution. It can be used for applications such as teaching, therapy planning, and so on where high accuracy is not essential.

ACKNOWLEDGEMENTS

This work was supported by the National Natural Science Foundation of China (grant nos. 61272298 and 61373084), Zhejiang Provincial Natural Science Foundation of China (grant no. Z1110154), the China 863 programme (grant no. 2012AA011503), and the Major Science and Technology Innovation Team (grant no. 2010R50040).

REFERENCES

1. Ji Z, Liu L, Wang Y. B-mesh: a modeling system for base meshes of 3D articulated shapes. *Computer Graphics Forum* 2010; **29**(7): 2169–2178.
2. Au O, Tai C, Chu H, Cohen-Or D, Lee T. Skeleton extraction by mesh contraction, In *ACM SIGGRAPH 2008, SIGGRAPH'08*, ACM, New York, NY, USA, 2008; 44:1–44:10.
3. Cao JJ, Tagliasacchi A, Olson M, Zhang H, Su ZX. Point cloud skeletons via Laplacian-based contraction, In *Proceedings of IEEE Conference on Shape Modeling and Applications, SMI'10*, IEEE Computer Society, Washington, DC, USA, 2010; 187–197.

4. Kanitsar A, Fleischmann D, Wegenkittl R, Sandner D, Felkel P, Gröller E. Computed tomography angiography: a case study of peripheral vessel investigation, In *Proceedings of the Conference on Visualization '01*, VIS '01, IEEE Computer Society, Washington, DC, USA, 2001; 477–480.
5. Oeltze S, Preim B. Visualization of anatomic tree structures with convolution surfaces, In *Proceedings of the Sixth Joint Eurographics—IEEE TCVG Conference on Visualization*, VISSYM'04, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 2004; 311–320.
6. Steffen O, Bernhard P. Visualization of vasculature with convolution surfaces: method, validation and evaluation. *IEEE Transactions on Medical Imaging* 2005; **24**(4): 540–548.
7. Zhu X, Guo X, Jin X. Efficient polygonization of tree trunks modeled by convolution surfaces. *SCIENCE CHINA Information Sciences* 2013; **56**(3): 1–12.
8. Hahn HK, Preim B, Selle D, Peitgen HO. Visualization and interaction techniques for the exploration of vascular structures, In *Proceedings of the Conference on Visualization*, VIS '01, IEEE Computer Society, Washington, DC, USA, 2001; 395–402.
9. Felkel P, Wegenkittl R, Buhler K. Surface models of tube trees, In *Proceedings of the Computer Graphics International*, CGI '04, IEEE Computer Society, Washington, DC, USA, 2004; 70–77.
10. Xu H, Gossett N, Chen B. Knowledge and heuristic-based modeling of laser-scanned trees. *ACM Transactions on Graphics* 2007; **26**(4): 19–31.
11. Livny Y, Yan F, Olson M, Chen B, Zhang H, El-sana J. Automatic reconstruction of tree skeletal structures from point clouds, In *ACM SIGGRAPH Asia 2010*, SIGGRAPH ASIA'10, ACM, New York, NY, USA, 2010; 151:1–151:8.
12. Pirk S, Niese T, Deussen O, Neubert B. Capturing and animating the morphogenesis of polygonal tree models. *ACM Transactions on Graphics* 2012; **31**(6): 169:1–169:10.
13. Pirk S, Stava O, Kratt J, Said MAM, Neubert B, Měch R, Benes B, Deussen O. Plastic trees: interactive self-adapting botanical tree models. *ACM Transactions on Graphics* 2012; **31**(4): 50:1–50:10.
14. Lluch J, Viv R, Monserrat C. Modelling tree structures using a single polygonal mesh. *Graphical Models* 2004; **66**(2): 89–101.
15. Ritter F, Hansen C, Dicken V, Konrad O, Preim B, Peitgen HO. Real-time illustration of vascular structures. *IEEE Transactions on Visualization and Computer Graphics* 2006; **12**(5): 877–884.
16. Pihuit A, Cani MP, Palombi O. Sketch-based modeling of vascular systems: a first step towards interactive teaching of anatomy, In *Proceedings of the Seventh Sketch-Based Interfaces and Modeling Symposium*, SBIM '10, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 2010; 151–158.
17. Tobler RF, Maierhofer S, Wilkie A. A multiresolution mesh generation approach for procedural definition of complex geometry, In *Proceedings of the Shape Modeling International 2002 (SMI'02)*, SMI '02, IEEE Computer Society, Washington, DC, USA, 2002; 35–43.
18. Klein J, Friman O, Hadwiger M, Preim B, Ritter F, Vilanova A, Zachmann G, Bartz D. Visual computing for medical diagnosis and treatment. *Journal of Computers and Graphics* 2009; **33**(4): 554–565.
19. Brentzen JA, Misztal MK, Wenicka K. Converting skeletal structures to quad dominant meshes. *Computers & Graphics* 2012; **36**(5): 555–561.
20. Catmull E, Clark J. Recursively generated b-spline surfaces on arbitrary topological meshes. *Computer Aided Design* 1978; **10**(6): 350–355.
21. Loop C. Smooth subdivision surfaces based on triangles, *Master's Thesis*, The University of Utah, Utah, USA, 1987.
22. Doo D, Sabin M. Behavior of recursive division surfaces near extraordinary points. *Computer Aided Design* 1978; **10**(6): 356–360.
23. Fabri A, Giezeman GJ, Kettner L, Schirra S, Schonherr S. On the design of CGAL a computational geometry algorithms library. *Software—Practice and Experience* 2000; **30**(11): 1167–1202.
24. Sovakar A, Kobbelt L. API design for adaptive subdivision schemes. *Computers & Graphics* 2004; **28**(1): 67–72.
25. Loop C, Schaefer S, Ni T, Castano I. Approximating subdivision surfaces with gregory patches for hardware tessellation. *ACM Transactions on Graphics* 2009; **28**(5): 1–9.
26. Xia J, Garcia I, He Y, Xin SQ, Patow G. Editable polycube map for GPU-based subdivision surfaces, In *Symposium on Interactive 3D Graphics and Games 2011*, I3D'11, ACM, New York, NY, USA, 2011; 151–158.
27. Patney A, Ebeida MS, Owens JD. Parallel view-dependent tessellation of Catmull–Clark subdivision surfaces, In *Proceedings of the Conference on High Performance Graphics 2009*, HPG'09, ACM, New York, NY, USA, 2009; 99–108.
28. Schwarz M, Stamminger M. Fast GPU-based adaptive tessellation with CUDA. *Computer Graphics Forum* 2009; **28**(2): 365–374.

29. Barthe L, Kobbelt L. Direct computation of a control vertex position on any subdivision level, In *IMA Conference on the Mathematics of Surfaces*, Lecture Notes in Computer Science, Springer: Berlin/Heidelberg, 2003; 40–47.
30. Stam J. Exact evaluation of Catmull–Clark subdivision surfaces at arbitrary parameter values, In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '98, ACM, New York, NY, USA, 1998; 395–404.
31. Nasri AH. Curve interpolation in recursively generated b-spline surfaces over arbitrary topology. *Computer Aided Geometric Design* 1997; **14**: 13–30.
32. Nasri AH. Recursive subdivision of polygonal complexes and its applications in computer-aided geometric design. *Computer Aided Geometric Design* 2000; **17**(7): 595–619.
33. Nasri AH, Abbas A. Lofted Catmull–Clark subdivision surfaces, In *Proceedings of Geometric Modeling and Processing*, GMP '02, Saitama, Japan, 2002; 83–93.
34. Nasri AH, Abbas AM. Designing Catmull–Clark subdivision surfaces with curve interpolation constraints. *Computers & Graphics* 2002; **26**: 393–400.
35. Nasri AH, Karam WB, Samavati F. Sketch-based subdivision models, In *Proceedings of the 6th Eurographics Symposium on Sketch-based Interfaces and Modeling*, SBIM '09, ACM, New York, NY, USA, 2009; 53–60.
36. Samavati FF, Bartels RH. Multiresolution curve and surface representation: reversing subdivision rules by least-squares data fitting. *Computer Graphics Forum* 1999; **18**(2): 97–119.

SUPPORTING INFORMATION

Supporting information may be found in the online version of this article.

AUTHORS' BIOGRAPHIES



Xiaoqiang Zhu is a faculty member of Shanghai University. He received his BSc degree in Mathematics in 2006 and MSc degree in Computer Science in 2009 from Hefei University of Technology and his PhD degree in Computer Science and Technology in 2013 from Zhejiang University. His research interests include implicit surface modeling and general-purpose graphics processing unit computing.



Xiaogang Jin is a professor of the State Key Lab of CAD&CG, Zhejiang University. He received his BSc degree in computer science in 1989, MSc and PhD degrees in applied mathematics in 1992 and 1995, all from Zhejiang University. His current research interests include implicit surface computing, special effects simulation, mesh fusion, texture synthesis, crowd animation, cloth animation, and facial animation.



Lihua You is currently an associate professor at the National Centre for Computer Animation, Bournemouth University, UK. He received his MSc degree and PhD degree from Chongqing University, China, and another PhD degree from Bournemouth University, UK. His current research interests are in computer graphics, computer animation, and geometric modelling. He has produced over 170 refereed publications and is a reviewer for over 30 international journals, some international conferences, and funding bodies.