COMPUTER ANIMATION AND VIRTUAL WORLDS
Comp. Anim. Virtual Worlds 2017; 28: e1683
Published online 30 November 2015 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/cav.1683

RESEARCH ARTICLE

A PDE-based head visualization method with CT data

Congkun Chen[1], Yun Sheng[1], Fang Li[2], Guixu Zhang[1]* and Hassan Ugail[3]

[1] The Department of Computer Science and Technology, School of Computer Science and Software Engineering, East China Normal University, Shanghai, China
[2] The Department of Mathematics, East China Normal University, Shanghai, 200062, China
[3] Centre for Visual Computing, University of Bradford, Bradford, UK

ABSTRACT

In this paper, we extend theuse of the partial differential equation (PDE) method to head visualization with computed tomography (CT) data and show how the two primary medical visualization means, surface reconstruction, and volume rendering can be integrated into one single framework through PDEs. Our scheme first performs head segmentation from CT slices using a variational approach, the output of which can be readily used for extraction of a small set of PDE boundary conditions. With the extracted boundary conditions, head surface reconstruction is then executed. Because only a few slices are used, our method can perform head surface reconstruction more efficiently in both computational time and storage cost than the widely used marching cubes algorithm. By elaborately introducing a third parameter w to the PDE method, a solid head can be created, based on which the head volume is subsequently rendered with 3D texture mapping. Instead of designing a transfer function, we associate the alpha value of texels of the 3D texture with the PDE parameter w through a linear transform. This association enables the production of a visually translucent head volume. The experimental results demonstrate the feasibility of the developed head visualization method. Copyright © 2015 John Wiley & Sons, Ltd.

KEYWORDS

PDE; surface reconstruction; solid modeling; volume rendering

***Correspondence**

Guixu Zhang, Department of Computer Science Technology, East China Normal University, Shanghai, China.
E-mail: gxzhang@cs.ecnu.edu.cn

1. INTRODUCTION

Bloor and Wilson pioneered geometric modeling with partial differential equations (PDEs) in computer graphics by proposing a parametric PDE-based surface blending approach two and a half decades ago [1]. Since then, advantages of the PDE method in geometric modeling have been gradually discovered and recognized. One principal advantage comes from the ability that the differential operators of PDEs ensure a generation of smooth surfaces, where the smoothness is strictly governed by the order of the PDEs used. This advantage has enabled the broad use of PDEs in a variety of geometric modeling issues, such as blend surface generation [1], interactive surface design [2], surface modeling [3,4], and solid modeling [5]. The other advantage of using the PDE method is that a PDE surface can be generated by intuitively manipulating a relatively small set of boundary conditions of the PDE, converting the geometric modeling issue into a PDE solving problem of boundary values. Therefore, the PDE method can be considered as a smoothing process in which a geometric model is obtained as a smooth transition between the boundary conditions. This property ensures the successful dabbling of the PDE method in computer-aided manufacturing [6,7], shape morphing [8], Web visualization [9], and mesh reconstruction [4], and so on.

In this paper, we propose to extend the use of the PDE method to head visualization with computed tomography (CT) data. The developed method allows for, under the same framework, the performance of both head surface reconstruction and volume rendering. Benefiting from the boundary value problem, our PDE method is subject to a contour-based method in surface reconstruction and can readily reconstruct the head surface using only a small group of boundary conditions. The boundary conditions are boundary curves that can be automatically extracted from 2D head CT slices with a variational segmentation method. Taking advantage of inherent parameterization, our method can generalize the reconstructed head surface model to a solid by elaborately introducing an extra third parameter, w, into the conventional uv parametric gird of PDEs. This third parameter not only can control the solidity of the head model but also can subsequently be employed to render the opacity of the head volume, producing a translucent visualization result. As the developed

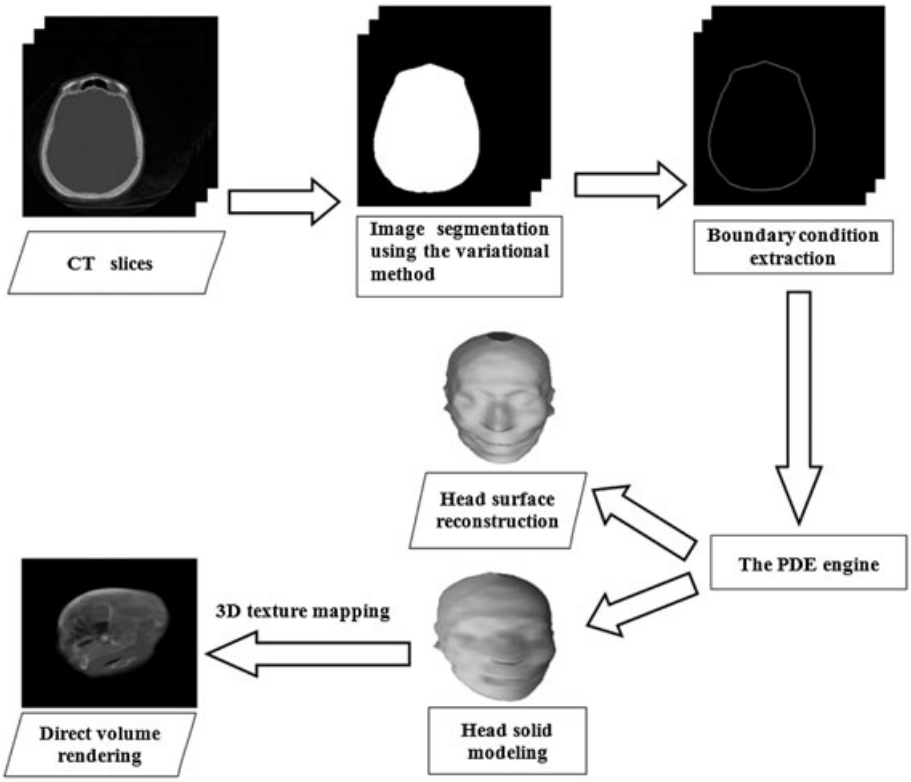


Figure 1. The flowchart of the partial differential equation (PDE) method for head visualization with computed tomography (CT) data.

volume rendering process is carried out by directly texture mapping the solid head model, it is subject to object-order volume rendering.

Surface reconstruction and volume rendering are the two primary technologies to visualize medical data. One of the most widely used surface reconstruction methods is the marching cubes (MC) algorithm, which is, however, time-consuming. For volume rendering, conventional methods require a well-designed transfer function, which assigns different objects of the volume data with different color and opacity values. However, an appropriate transfer function is complicated to design. There also exist open-source software libraries for medical data analysis and visualization, such as 3D Slicer and FMRIB Software Library, using embedded conventional methods for visualization. Nevertheless, neither 3D Slicer nor FMRIB Software Library supports both surface reconstruction and volume rendering under the same framework. Compared with the MC algorithm, our PDE method is more efficient in both computational time and storage cost, as it uses only a small number of boundary conditions. Because of the parametric form, our PDE method allows for control of the levels of detail of the reconstructed head model by adjusting its parametric resolution. Instead of using the transfer function, our method elaborately associates the opacity value of the volume with the PDE parameter w in a linear manner to produce translucent volume rendering results, achieving visually pleasant head visualization results by integrating both surface reconstruction and volume rendering into one single framework through PDEs.

Figure 1 shows the flowchart of the developed method for head visualization with CT slices. Given a small number of slices manually chosen from a sequence of head cross-sectional slices, image segmentation is first carried out ahead of boundary curve extraction. The variational method is utilized in segmentation because it is robust to noise and able to generate relatively precise segmentation results. Then, the extracted boundary curves are input to the PDE engine as boundary conditions. These boundary conditions should contain as much as possible the necessary geometry of the head in order to reconstruct the head precisely. Either the head surface or the head solid model can be reconstructed by solving a group of PDEs, where each PDE produces one unique PDE patch, and one head model may consist of several PDE patches. Following head solid modeling, the head volume is rendered by mapping a 3D texture into the reconstructed solid grid.

The rest of the paper is structured as follows. The background knowledge upon medical data visualization and the PDE methods are briefed in Section 2. In Section 3, we introduce the extraction process of boundary conditions. In Section 4, we elaborate the developed head visualization method, consisting of head surface reconstruction, head

DOI: 10.1002/cav

solid modeling, and volume rendering. In Section 5, experimental results are given to justify the developed method. Section 6 concludes the paper.

2. BACKGROUND

Head visualization has played a crucial role in many medical fields, where medical imaging technologies, such as CT and magnetic resonance imaging (MRI), have been widely used to scan the human head. These medical imaging technologies produce only 2D parallel slices without a 3D illusion. Head visualization can synthesize the 3D head geometry by making use of the information presented in 2D head medical slices, thus providing doctors a more realistic and intuitive diagnostic basis. Moreover, in forensic medicine, head visualization recovers the unknown facial features of an individual with head medical data for recognition and identification [10,11]. In archaeology, archaeologists need the head visualization technology to recover the authentic facial features of historical figures [12,13]. Besides, head visualization was also used to simulate facial aesthetic surgery [14,15]. Because the mainstream head visualization methods stem from the medical data ones, this section briefs state of the art in medical data visualization, summarizes the existing PDE methods, and elaborates the Bloor–Wilson PDE (BWPDE) method adopted in the paper.

2.1. Medical Data Visualization

Medical data visualization technologies can be divided into multiplanar rendering, surface rendering, and volume rendering [16]. Multiplanar rendering visualizes gray values of an arbitrary cross section of volumetric data. The axial slices are cut into different orthogonal planes, allowing to view the entire structure orthogonally. The rendered data are also 2D as original slices and thus gives no more insight into the 3D structure than the original ones.

Surface reconstruction algorithms can be classified into contour-based surface reconstruction and iso-surface extraction [16]. Contour-based surface reconstruction methods [17] need to extract the cross-sectional contour of each axial slice and then reconstruct the 3D surface through triangulation between the consecutive contours. When there exist more than one contour in an axial slice, ambiguity in connecting consecutive contours may arise. For iso-surface extraction, the most widely used algorithm is the MC algorithm [18], proposed by Lorensen and Cline in 1987. But there also exists an ambiguity problem during triangulation of the MC algorithm. Marching tetrahedrons [19] were then proposed to overcome the ambiguity problem of the MC algorithm. The marching tetrahedrons algorithm splits each cube into six tetrahedrons by cutting diagonally through each of the three pairs of opposite faces and creates the iso-surface based on these tetrahedrons rather than cubes. Another alternative for MC is dividing cubes [20], where cubes are divided into pixelwise sub-cubes. All of these algorithms can produce reasonable surface reconstruction results, but they are computationally complex.

The mainstream volume rendering methods can be classified into two categories: the image-order and object-order rendering methods. The main difference between the image-order and object-order methods is that the former renders from the viewport of the imaging plane while the latter from that of volume data. Ray tracing [21,22] and maximum intensity projection [23] are subject to the image-order volume rendering algorithms, because they both cast parallel rays from each pixel of the imaging plane. The difference between these two methods is that, the ray tracing algorithm makes all the voxels along a ray contribute to the projection pixel, while the maximum intensity projection algorithm picks only the voxel with a maximum intensity value along a ray. In contrast to the image-order rendering algorithms, the object-order methods, such as shear warp factorization [24] and texture-based volume rendering [25], render the volume from the viewport of volume data. The shear warp factorization algorithm was proposed to reduce the computation cost by factoring the view transformation, while the texture-based approach performs the rendering process by applying 2D textures to the geometric primitives, such as quadrangles.

However, the aforementioned volume rendering approaches disregard optical properties observed from real translucent materials. To this end, an improvement based on shading model has been proposed, allowing for sophisticated global illumination [26]. Moreover, with the advancement in graphics processing unit (GPU), GPU-based methods have been proposed to accelerate the rendering process [27,28], and state of the art in GPU-based rendering methods can be found in [29]. In order to make rendering results visually translucent and noiseless, the aforementioned volume rendering methods adopt a transfer function [30]. However, an appropriate transfer function is complicated to design. A survey of these illumination techniques are reviewed in [31]. Instead of designing a complicated transfer function, our PDE-based method implements translucent and noiseless rendering by subtly relating the alpha value to one of the PDE parameters before 3D texturing the head solid model.

In order to deal with the occlusion problem in visualization, the technique of magic lens was proposed to change the rendering styles of desired parts during visualization. In this way, magic lens makes it possible to specify the parts to be rendered in different styles. In [32], Kirmizibayrak *et al*. have extended the magic lens to handle volume editing operations by using the lens as a volumetric brush.

Moreover, different medical imaging devices measure different properties of the same region. There exist demands for visualization of multi-modality data, dealing simultaneously with various datasets, such as CT and MRI data, in order to provide a better perception of anatomical structures [33].

Besides, because of the improvement of hardware in mobile devices, visualization on mobile devices is a promising research direction. Nevertheless, because mobile devices have limited data processing ability, visualization on mobile devices is a challenging task. Researchers have studies the possibility of visualization on mobile devices. For example, Noguera *et al*. proposed a volume rendering strategy on mobile devices based on texture mapping [34]. Because our PDE-based method uses only a small amount of boundary conditions, it is possible to apply our PDE method to visualization on mobile devices.

2.2. The Partial Differential Methods

The mainstream PDE methods in geometric modeling are summarized in Table I with the order of PDEs used, as well as their solution methods and relevant applications.

The major distinctions among the methods listed in Table I are the following:

- Solutions to these PDE methods are different. Theoretically, the PDE method can be solved either numerically or analytically. An analytic solution is for those closed boundaries; otherwise, a numerical method is sought after. However, the numerical method is more computationally expensive.
- Different PDEs are chosen in different circumstances. The BWPDE method employs the biharmonic-like fourth-order PDE. However, it can only ensure a C^1 continuity between PDE patches. If C^2 continuity is demanded, a sixth-order PDE [35] has to be used, leading to a higher computational cost. Moreover, it has been proved that the BWPDE method can be extended to an arbitrary order allowing complex shape design with a single PDE [36].

Among the aforementioned three PDE methods, the BWPDE method has been the most computationally efficient with a widest range of applications.

2.3. The Bloor–Wilson Partial Differential Method

The PDE method treats surface generation as a solution to a specific PDE subject to certain boundary conditions. The reason for using the fourth-order PDE is that a lower-order PDE has no freedom to specify the reconstructed surface smoothness, whereas the calculation of the higher-order PDE is more time-consuming. In order to reach a trade-off between the surface smoothness and computational complexity, a biharmonic-like fourth-order PDE is chosen:

$$\left(\frac{\partial^2}{\partial u^2} + a^2\frac{\partial^2}{\partial v^2}\right)^2 \mathbf{X}(u,v) = 0 \tag{1}$$

where $\mathbf{X}(u,v)$ represents the 3D surface subject to a 2D parametric uv domain with the range $0 \le u \le 1,\ 0 \le v \le 2\pi$, which has the following form:

$$\mathbf{X}(u,v) = (x(u,v), y(u,v), z(u,v)) \tag{2}$$

The parameter a is called the smoothing parameter, governing the relative rate of smoothing between the u and v directions. To solve (1), four boundary conditions are needed:

$$\mathbf{X}(0,v) = \mathbf{P}_0(v) \tag{3}$$

$$\mathbf{X}(p_1,v) = \mathbf{P}_1(v) \tag{4}$$

$$\mathbf{X}(p_2,v) = \mathbf{P}_2(v) \tag{5}$$

$$\mathbf{X}(1,v) = \mathbf{P}_3(v) \tag{6}$$

$\mathbf{P}_0(v)$, $\mathbf{P}_1(v)$, $\mathbf{P}_2(v)$, and $\mathbf{P}_3(v)$ represent the boundary conditions when $u = 0$, p_1, p_2, and 1, respectively, and $0 < p_1 < p_2 < 1$.

Imposing an analytic solution, (1) can be solved by separation of variables as follows:

Table I. The three mainstream PDE methods.

Methods	The order of PDEs	Solutions	Applications
Bloor–Wilson	Fourth, sixth, and *n*th	Fourier analysis (analytical method)	Blend surface [1], free-form surface [3], CAM [6,7], shape morphing [8], facial modeling [37], interactive design [36], and mesh approximation [4]
Zhang and You	Fourth and sixth	Pseudo-levy series (analytical method)	Blend surface [38] and free-form surface [39]
Du and Qin	Sixth	Finite difference (numerical method)	Free-form surface [40] and solid modeling [41]

PDE, partial differential equation; CAM, computer-aided manufacturing.

DOI: 10.1002/cav

$$\mathbf{X}(u,v) = \mathbf{A}_0(u) + \sum_{n=1}^{\infty} [\mathbf{A}_n(u)\cos(nv) + \mathbf{B}_n(u)\sin(nv)] \tag{7}$$

where $\mathbf{A}_n(u)$ and $\mathbf{B}_n(u)$ are of the following forms:

$$\mathbf{A}_0(u) = \mathbf{a}_{00} + \mathbf{a}_{01}u + \mathbf{a}_{02}u^2 + \mathbf{a}_{03}u^3 \tag{8}$$

$$\mathbf{A}_n(u) = (\mathbf{a}_{n1} + \mathbf{a}_{n2}u)e^{anu} + (\mathbf{a}_{n3} + \mathbf{a}_{n4}u)e^{-anu} \tag{9}$$

$$\mathbf{B}_n(u) = (\mathbf{b}_{n1} + \mathbf{b}_{n2}u)e^{anu} + (\mathbf{b}_{n3} + \mathbf{b}_{n4}u)e^{-anu} \tag{10}$$

where the vector-valued coefficients $\mathbf{a}_{00}$, $\mathbf{a}_{01}$, $\mathbf{a}_{02}$, $\mathbf{a}_{03}$, $\mathbf{a}_{n1}$, $\mathbf{a}_{n2}$, $\mathbf{a}_{n3}$, $\mathbf{a}_{n4}$, $\mathbf{b}_{n1}$, $\mathbf{b}_{n2}$, $\mathbf{b}_{n3}$, and $\mathbf{b}_{n4}$ in (8)–(10) can be determined by the four boundary conditions defined by (3)–(6).

In (7), $\mathbf{A}_0(u)$ is considered to be the "spine" of the reconstructed surface, which takes the form of a cubic polynomial curve with respect to u, while the remaining terms represent a summation of "radius" vectors that give the position of reconstructed surface $\mathbf{X}(u,v)$ relative to the "spine." As a result, the PDE surface patch may be pictured as a sum of the spine vector $\mathbf{A}_0(u)$, plus a primary radius vector $\mathbf{A}_1(u)\cos(v) + \mathbf{B}_1(u)\sin(v)$, plus a secondary radius vector $\mathbf{A}_2(u)\cos(2v) + \mathbf{B}_2(u)\sin(2v)$ attached to the end of the primary radius, and so on. The amplitude of the radius term decays as the frequency increases. It can be observed that the first few radii contain the most essential geometric information. Thus, (7) can be further approximated by the sum of a finite number of Fourier modes plus a reminder term, $\mathbf{R}(u,v)$:

$$\mathbf{X}(u,v) = \mathbf{A}_0(u) + \sum_{n=1}^{N} [\mathbf{A}_n(u)\cos(nv) + \mathbf{B}_n(u)\sin(nv)] + \mathbf{R}(u,v) \tag{11}$$

where

$$\mathbf{R}(u,v) = [\mathbf{r}_1(v) + \mathbf{r}_2(v)u]e^{\omega u} + [\mathbf{r}_3(v) + \mathbf{r}_4(v)u]e^{-\omega u} \tag{12}$$

and $\omega = a(N+1), \mathbf{r}_1(v), \mathbf{r}_2(v), \mathbf{r}_3(v), \mathbf{r}_4(v)$ are functions that can be determined by calculating the difference between the original boundary conditions and the ones satisfied by $\mathbf{F}(u,v)$:

$$\mathbf{F}(u,v) = \mathbf{A}_0(u) + \sum_{n=1}^{N} [\mathbf{A}_n(u)\cos(nv) + \mathbf{B}_n(u)\sin(nv)] \tag{13}$$

An elaborate illustration of the PDE method can be found in [42]. Examples of free-form surface generation with the PDE method are shown in Figure 2.

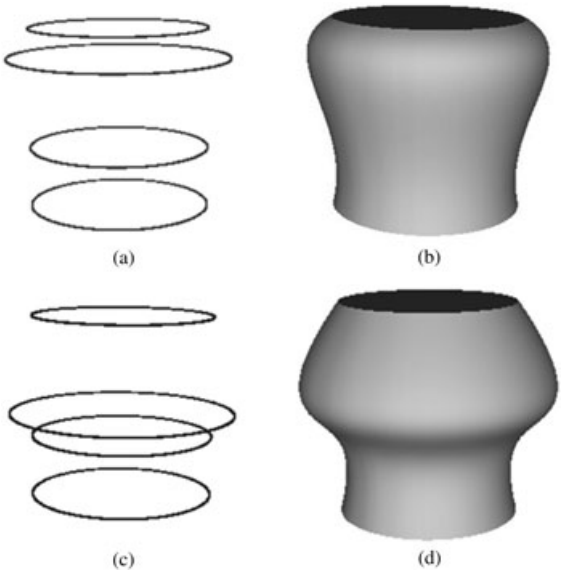


Figure 2. Examples of free-form surfaces generated by applying different boundary conditions. (a and c) Different boundary conditions; the difference between (a) and (c) is that (c) is formed by translating the second boundary of (a) downward. (b and d) The free-form surfaces generated using the partial differential equation method satisfying the boundary conditions shown in (a) and (c), respectively.

3. BOUNDARY EXTRACTION

This section is dedicated to the extraction process of PDE boundary conditions, consisting of medical image segmentation and boundary condition extraction. Medical image segmentation that determines the accuracy of boundary condition extraction plays a key role in our PDE-based visualization framework. A precise segmentation of the head region from cross-sectional medical slices guarantees realistic reconstruction and visualization. Currently, the mainstream medical image segmentation methods include the threshold methods, edge-based methods, region-based methods, and special theory-based methods. The image segmentation method adopted in this paper belongs to the special theory-based category, called variational image segmentation [43,44]. The main advantages of variational image segmentation against other methods are that it is robust to noise and can guarantee the segmentation accuracy.

In 2001, Chan and Vese [44] proposed a segmentation method based on the active contour without edges. The method minimizes

$$\min_{\phi,c_1,c_2} \left\{ \int_{\Omega} |\nabla H(\phi)|\,\mathrm{d}x + \lambda \int_{\Omega} |I - c_1|^2 H(\phi)\mathrm{d}x + \int_{\Omega} |I - c_2|^2 (1 - H(\phi))\mathrm{d}x \right\} \tag{14}$$

where $I(x) : \Omega \to R$ is the given image, $\Omega \subset R^2$ is a

rectangle region, $\phi(x) : \Omega \rightarrow R$ is the level set function, $H(\phi)$ is the Heaviside function satisfying

$$H(\phi) = \begin{cases} 1, & \phi \geq 0 \\ 0, & \phi < 0 \end{cases} \tag{15}$$

c_1 and c_2 are constants, $\lambda > 0$ is a fixed parameter, and $x = (x_1, x_2)$ is the coordinate of Ω. The final segmentation curve is given by $\phi = 0$. $\phi \geq 0$ and $\phi < 0$ represent the two regions segmented by the curve at $\phi = 0$. To solve (14), a negative gradient descent method in [44] is usually used, which, however, converges slowly because of the small size of the time step.

For the sake of efficiency, we use a soft segmentation model. We generalize the model for medical data segmentation, called 3D Chan–Vese soft segmentation model. This is because in medical image segmentation, a sequence of 2D CT or MRI slices, if consecutive scans are close enough, can be considered as 3D data. We have the volume data $I(x) : \Omega \subset R^3 \rightarrow R$. The 3D Chan–Vese soft segmentation model can be written as the following energy minimization problem:

$$\begin{aligned} \min_{p\in[0,1],c_1,c_2} \Big\{ E(p,c_1,c_2) &= \int_\Omega |\nabla p| \, \mathrm{dx} \\ &+ \lambda \int_\Omega |I - c_1|^2 p \mathrm{dx} \\ &+ \lambda \int_\Omega |I - c_2|^2 (1-p) \, \mathrm{dx} \Big\} \end{aligned} \tag{16}$$

Here, we use a 3D soft membership function $p(x) \in [0, 1]$ to replace the 2D hard segmentation function $H(\phi(x)) \in \{0, 1\}$ in (14); note that c_1 and c_2 remain constant.

To solve (16), we make use of the alternating minimization method and Chambolle's fast dual-projection method. Firstly, we fix p and solve c_1 and c_2. Let $\frac{\partial E}{\partial c_1} = 0$, $\frac{\partial E}{\partial c_2} = 0$, and we can easily deduce

$$\begin{cases} c_1 = \frac{\int_\Omega I p \mathrm{dx}}{\int_\Omega p \mathrm{dx}} \\ c_2 = \frac{\int_\Omega I(1-p) \mathrm{dx}}{\int_\Omega (1-p) \mathrm{dx}} \end{cases} \tag{17}$$

Then we fix c_1 and c_2 and solve p. The sub-problem can be simplified as

$$\min_{p\in[0,1]} \left\{ \int_\Omega |\nabla p| \, \mathrm{dx} + \lambda \int_\Omega r p \mathrm{dx} \right\} \tag{18}$$

where $r = |I - c_1|^2 - |I - c_2|^2$. It is difficult to solve (18) by a direct method because the first term (total variation) is not differentiable. Thus, we add a new auxiliary variable q and then relax (18) as

$$\min_{p,q\in[0,1]} \left\{ \int_\Omega |\nabla p| \, \mathrm{dx} + \lambda \int_\Omega r q \mathrm{dx} + \frac{\mu}{2} \int_\Omega (p-q)^2 \mathrm{dx} \right\} \tag{19}$$

where the last term requires that p and q should be close enough. Then we use the alternating minimization again to solve (19). By fixing p, the sub-problem of q is a convex constrained optimization problem. It is straightforward to derive the solution in closed form:

$$q = \min(\max(p - \theta r, 0), 1) \tag{20}$$

where $\theta = \frac{\lambda}{\mu}$. By fixing q, the sub-problem of p is a standard Rudin–Osher–Fatemi model [45]. We can use Chambolle's fast dual-projection method [46] to solve it. The iteration formulas are

$$\begin{cases} p = q - \theta \mathrm{div}(s) \\ s = \frac{s + \tau(\nabla(\mathrm{div}(s) - q/\lambda))}{1 + |\tau(\nabla(\mathrm{div}(s) - q/\lambda))|} \end{cases} \tag{21}$$

where s is the dual variable of p and τ is the step size. Hence, the method to solve (16) becomes to alternately iterate (17), (20), and (21). Our method is efficient, and generally 30 iterations are enough to obtain a satisfactory solution.

With four head CT slices arbitrarily chosen from the dataset [47], we use the introduced algorithm to segment the head. The parameters are set as follows: $\lambda = 0.003, \theta = 0.1, \tau = 0.25$. We remark that the algorithm is not sensitive to θ and τ. λ controls the smoothness of the segmentation (the smaller the smoother). It can be seen in Figure 3 that the developed 3D Chan–Vese soft segmentation algorithm results in some holes. For example, some tissues inside the head are missed out. To tackle this problem, the largest connected component is extracted by removing all the other small components with the area less than some threshold and filling up the holes. The third row of Figure 3 shows the final segmentation results. Following segmentation, a morphological algorithm is performed to extract boundary conditions:

$$\beta(I) = I - (I \ominus S) \tag{22}$$

where S is a specific structuring element, I is the original image, $\beta(I)$ denotes the object boundary, and $\ominus$ is the morphological erosion operation. The extraction results of boundary conditions based on the image segmentation results are highlighted in red in the fourth row of Figure 3.

4. HEAD VISUALIZATION

This section describes in detail the head visualization process in our PDE method, where head surface reconstruction, head solid modeling, and head volume rendering are in turn performed.

4.1. Head Surface Reconstruction

After boundary condition extraction, the head surface can be readily reconstructed by solving a group of PDEs with these extracted boundary conditions. Because the PDE

DOI: 10.1002/cav

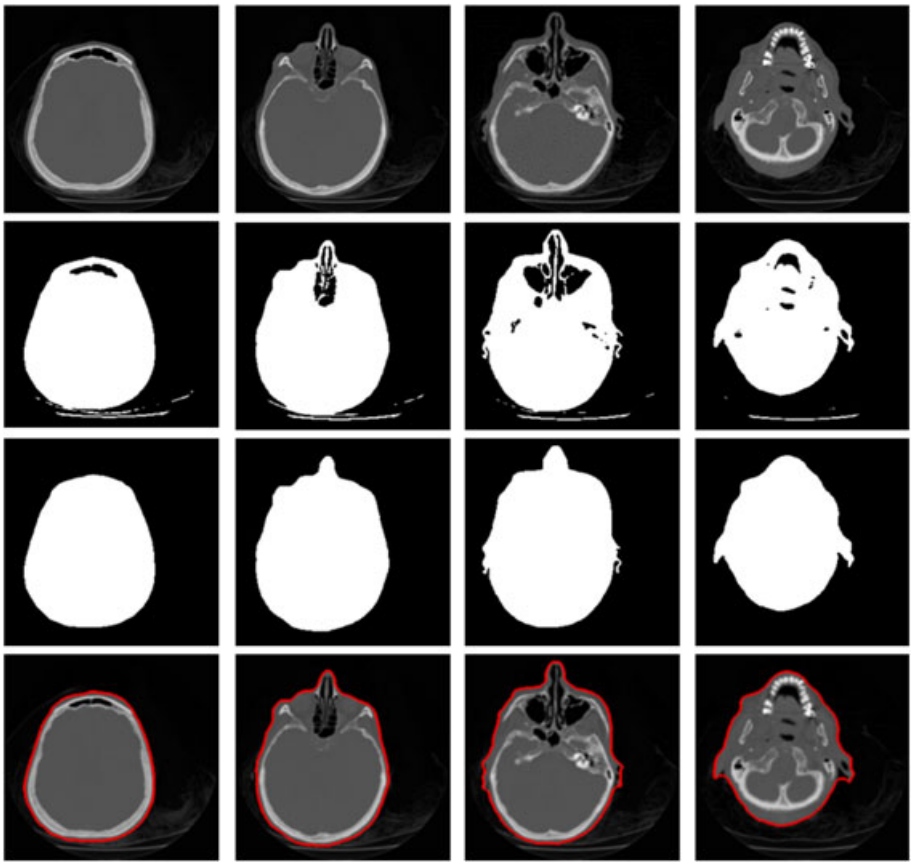


Figure 3. Medical image segmentation and boundary extraction. The first row shows four original computed tomography slices arbitrarily selected from the dataset. The second row shows the segmentation results using the developed method. The third row shows the maximum connected component of the second row. The fourth row shows the boundaries (red curves) extracted from the segmented images in the third row.

method can be regarded as a smooth interpolation between given boundary conditions, it is unnecessary to use up all the extracted boundary conditions for head reconstruction. In other words, for the sake of computational efficiency, it is unnecessary to deal with all the head CT slices, especially for those adjacent slices with similar geometry. Taking this advantage of the PDE method, we can use a relatively small number of PDEs for head visualization. The number of the PDEs required may vary on demand, and each PDE only produces a unique PDE patch for the head surface. Because of its geometric complexity, the whole head surface consists of several PDE patches in order to explicitly cover all the facial organs, such as the ears, nose, mouse, and eyes.

Suppose that we need t PDE patches to represent the whole head surface. With the fourth-order PDEs, we require four boundary conditions for generating each PDE patch. If any two adjacent patches share one common boundary condition, then a total of $(3t+1)$ boundary conditions are needed and $(3t+1)$ slices are selected from the dataset. The precision of the reconstructed head surface varies, dependending on the value of t. Intuitively speaking, as t increases, the more medical slices are involved and the more geometric detail of the head can be reconstructed.

The whole parametric domain is within the following range: $0 \leq u \leq 1$, $0 \leq v \leq 2\pi$. Because the whole head surface is partitioned into t PDE patches along the u direction, each PDE patch accounts for $1/t$ of u with v remaining from 0 to 2π. As shown in Figure 4, the whole head surface is divided into 10 PDE patches, that is, a total of 31 boundary conditions are needed in Figure 4(a). Figure 4(b) shows a 40×60 uv parametric grid. The results shown in Figure 4(c) and (d) are produced by the PDE method subject to the boundary conditions in Figure 4(a) and the resolution of the parametric domain in Figure 4(b).

Because the PDE method adopted in this paper is a parametric representation scheme, the head surface can be reconstructed in multiresolution by simply scaling down or up the resolution of the parametric uv domain. Therefore, the geometric detail of the reconstructed head surface is controllable by adjusting the resolution of its parametric domain, without increasing the number of PDE patches involved. This advantage of the developed method may

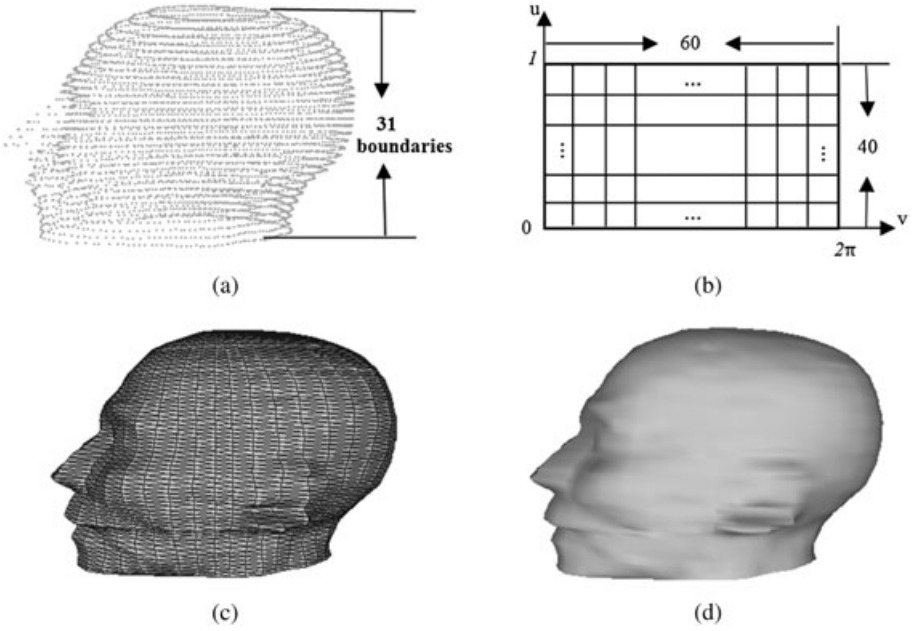


Figure 4. Head surface reconstruction using the partial differential equation method, with $0 \leq u \leq 1, 0 \leq v \leq 2\pi$. (a) The discreted boundary curves. (b) The sampling in the *uv* parametric domain. (c) The head surface in flat view. (d) The head surface in shaded view.

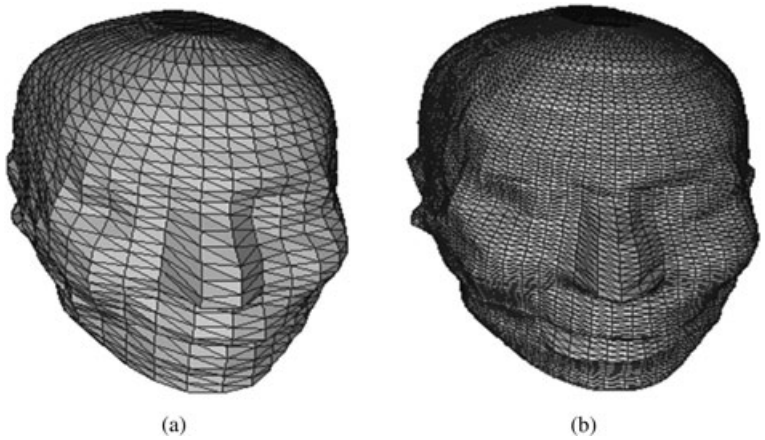


Figure 5. Multiresolutional reconstruction using the partial differential equation method, with $0 \leq u \leq 1, 0 \leq v \leq 2\pi$. (a) The head surface in 30 × 40. (b) The head surface in 80 × 80.

help save a considerable amount of storage expense and facilitate the computation. Such multiresolutional heads are illustrated in Figure 5.

4.2. Head Solid Modeling

A solid model is the geometric representation of a physical solid object. Solid modeling has a variety of applications from computer-aided design to health care. The common methods to create a solid model include constructive solid geometry, boundary representation, and cell decomposition. Du and Qin [40] proposed to represent a solid object with a trivariate parametric PDE method, where boundary surfaces or boundary curve networks are taken as the PDE boundary conditions. In this paper, the PDE method is extended to solid modeling by only introducing a third parameter w to its solution (Equation (7)):

$$\mathbf{X}(u,v,w) = \mathbf{A}_0(u) + w\sum_{n=1}^{\infty}[\mathbf{A}_n(u)\cos(nv) + \mathbf{B}_n(u)\sin(nv)] \tag{23}$$

where $0 \leq w \leq 1$. Geometrically, the term $\mathbf{A}_0(u)$ is regarded as the spine of the solid and also known as the medial axis or the skeleton. The term $\sum_{n=1}^{\infty}[\mathbf{A}_n(u)\cos(nv) + \mathbf{B}_n(u)\sin(nv)]$ specifies the distance from the points on

DOI: 10.1002/cav

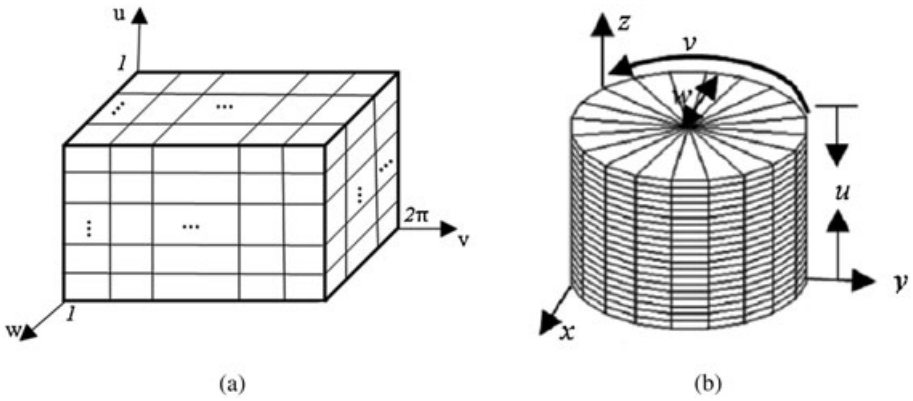


Figure 6. A mapping from the parametric domain to the physical domain. (a) The parametric domain. (b) The physical domain.

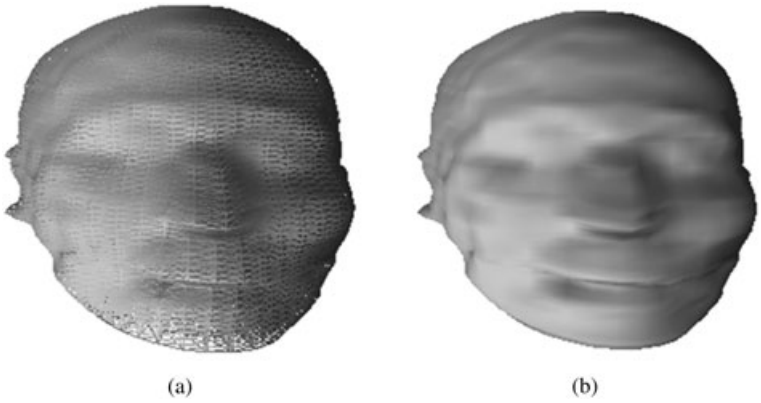


Figure 7. The head solid illustration using the partial differential equation method with $0 \leq u \leq 1, 0 \leq v \leq 2\pi, 0 \leq w \leq 1$. (a) The solid head in grid view. (b) The solid head in shaded view.

the surface $\mathbf{X}(u, v, 1)$ to the spine. Thus, by sampling w between 0 and 1 over u and v, a solid grid can be created from the spine towards its surface according to the given boundary conditions. In this way, our PDE method can define a solid model without changing the boundary conditions used for head surface reconstruction, and it can automatically recover the interior information in the solid model. Figure 6 depicts such a mapping from the parametric domain formed by u, v, and w to the physical domain. Figure 7 shows the head solid model generated by our PDE method, where the whole head model is made up of 10 PDE patches with 31 boundary conditions as shown in Figure 4(a).

4.3. Volume Rendering of the Head

Most volume rendering methods require a well-designed transfer function in order to make the rendering result visually translucent and noiseless. However, the difficulty still remains in transfer function design. In this section, we introduce how to obtain an analogous rendering effect through our PDE method.

4.3.1. Three-dimensional Texture Mapping.

An intuitive means for direct volume rendering is texture mapping, where all the given 2D medical slices are stacked up to form a 3D texture map, and the 3D texture is mapped to a series of pre-aligned parallel planes. One disadvantage of this method is that the volume-rendered result may contain noise if there is noise in the given medical slices. For example, Figure 8(a) shows an intuitively rendered volume of the head with the noise from the original medical data.

Instead, we propose to map the 3D texture to the solid grid reconstructed by the PDE method. In order to perform the mapping, texels corresponding to the vertices of the solid model need to be worked out. The correspondence between the texels and vertex coordinates can be found by normalizing the vertex coordinates in the x, y, and z directions. Because the solid grid is reconstructed by the PDE method with the extracted boundary conditions, the noise existing outside the head boundaries in the cross-sectional slices can be readily removed after rendering, as shown in Figure 8(b).

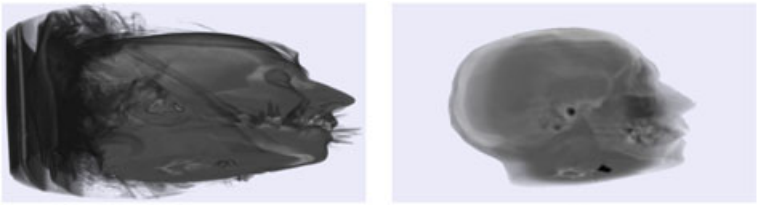


Figure 8. The volume rendering results. (a) The rendering result by intuitive texture-mapping method with alpha values associated with their corresponding pixel values. (b) The rendering result by our partial differential equation method.

4.3.2. Linear Interpolation of Alpha Values.

Alpha blending plays a crucial role in head volume rendering, where the alpha values of texels ranging between 0 and 1 need to be specified. The alpha value determines the opacity of the rendered volume. The larger the alpha value, the higher the volume opacity. An intuitive way to set the alpha value of each texel is to associate it with its corresponding pixel value in the slice image. However, this cannot achieve a fully perspective visualization effect, as shown in Figure 8(a).

To obtain rendering results in high quality, a transfer function is designed in conventional volume rendering algorithms. The transfer function maps volume data to the optical quantities, such as opacity and color. A well-designed transfer function reveals the structure of the volume data with some degree of details. However, an appropriate transfer function is complicated to design. In this paper, in order to avoid the complicated transfer function design, we simply relate the alpha value of the volume data with the parameter w in a linear manner.

In our method, because the parameter w in the parametric domain varies from 0, the spine of the solid model, to 1, the outermost iso-surface, we can take advantage of this property of the PDE method by linearly associating the alpha value with the parameter w:

$$\alpha = (l_\alpha - h_\alpha)\, w + h_\alpha \tag{24}$$

where l_α and h_α indicate the lowest and highest α values, respectively. This means that the outermost iso-surface possesses the highest transparency and the transparency becomes lower as the iso-surface goes inward, thus leading to a gradually perspective and translucent visualization result, as shown in Figure 8(b).

5. EXPERIMENTAL RESULTS

This section is dedicated to experiments of the developed head visualization method. The test data used in the experiments are acquired from [47–49]. Reference [47] offers CT head data containing 99 slices, while the head data from [48,49] are in Digital Imaging and Communications in Medicine, a standard format in medical imaging (DICOM) format containing 187 and 221 slices, respectively. All the tested medical slices are 256×256 in size.

5.1. Boundary Extraction Evaluation

Because our PDE method is a boundary value problem, the precision of boundary extraction plays a crucial role in the whole visualization method. In this paper, we evaluate our boundary extraction results in accordance with the ground truth. As shown in Figure 9, the regions enclosed by the white curves were segmented manually as the ground truth, while the regions enclosed by the red curves were segmented by our method. It can be observed that the extracted boundaries by our method overlap well with the ground truth.

The segmentation accuracy is evaluated by the Jaccard (JC) measure in this paper. The JC measure reported in [50] is a common measure that evaluates the segmentation accuracy based on regional overlaps. The value of the JC measure ranges between 0 and 1. When the value approaches 1, a high degree of overlap occurs between the segmentation result and the ground truth, implying a high segmentation accuracy. The evaluation result of the dataset in [47] is plotted in Figure 10, where the average segmentation accuracy is 0.9815. It is noticeable that there are valleys in the curve shown in Figure 10, which imply the relatively inaccurate cases underneath. These inaccuracies

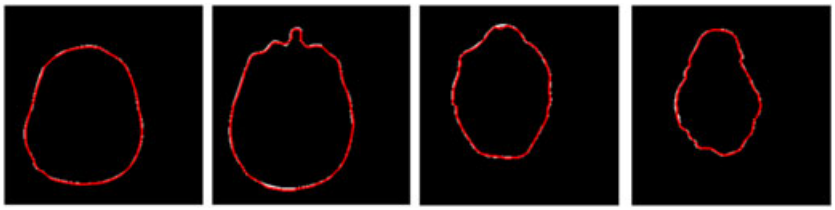


Figure 9. The comparison between the extracted boundaries and the ground truth. The red curves indicate the boundaries extracted from the segmentation results, while the white curves indicate the ground truth.

DOI: 10.1002/cav

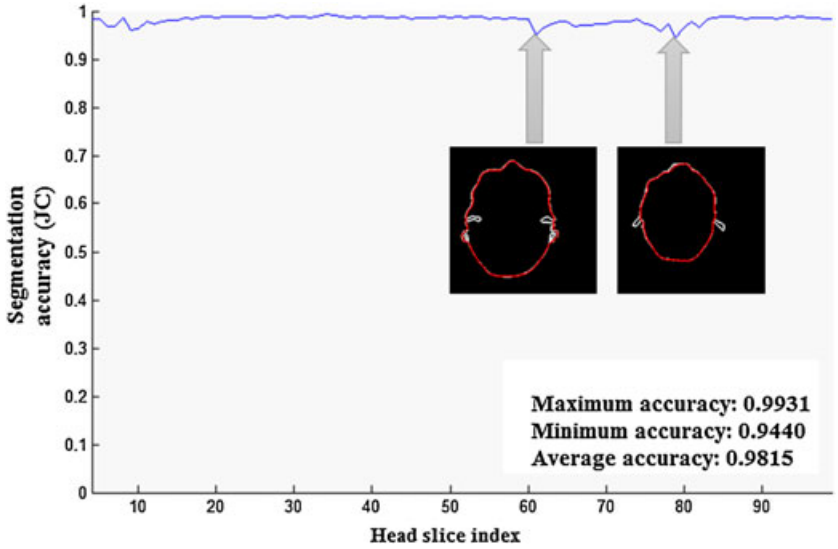


Figure 10. Segmentation accuracy measurement with the Jaccard (JC) measure for the dataset in [47].

Table II. Segmentation accuracy measurement with the JC measure on different medical data.

Dataset	Resolution	Minimum accuracy	Maximum accuracy	Average accuracy
Cthead [47]	256 × 256 × 99	0.9440	0.9931	0.9815
VHF-Head [48]	256 × 256 × 187	0.9623	0.9873	0.9806
Neuro-Axial [49]	256 × 256 × 221	0.9543	0.9899	0.9791

are caused mainly by the morphological operation introduced in boundary condition extraction. The segmentation evaluation results of three head medical datasets shown in Table II indicate a high degree of overlaps between the segmentation results and the ground truth, demonstrating the feasibility of the variational segmentation approach proposed in this paper.

5.2. Head Surface Reconstruction Results

Head surface reconstruction has been tested by comparing our results with those produced by the MC algorithm. Figure 11(a) and (b) shows the shaded head surface models reconstructed by our PDE method with t set to 10, that is, a total of 31 slices are manually selected from the dataset. Figure 11(c) and (d) shows the results produced by the MC algorithm using up all the 99 slices in the dataset. As we can see, the head surface models reconstructed by our PDE method appear close to those produced by the MC algorithm, with all the facial features well retained, although only a small number of head medical slices are used. Moreover, the noise around the mouth existing in the medical data, present in both Figure 11(c) and (d), has been suppressed in our results in Figure 11(a) and (b).

Comparisons in computational time and storage efficiency between the MC algorithm and our PDE method have also been performed. The experiments were carried out with Pentium(R) Dual-Core CPU 3.20 GHz and 4.00-GB RAM. Our PDE method took only 0.66 seconds, while the MC algorithm consumed 874.75 seconds for constructing the models in Figure 11. Note that the time consumed by our PDE method excludes boundary extraction. However, for 10 PDE patches, we need 31 medical slices, the boundary curves of which are extracted by the proposed method with 14.23 seconds. Even then, the head surface reconstruction method in this paper still outperforms well the MC algorithm in terms of computational efficiency. In storage cost, the MC algorithm uses all slices in the dataset, and the surface model reconstructed by the MC algorithm is 37 647 kB in size. In contrast to the MC algorithm, rather than storing the whole reconstructed head surface model, we can choose to store only the PDE boundary conditions and reconstruct the head surface model by solving the PDEs when needed. For 31 boundary conditions, and each boundary condition is about 2 kB in size, only 62 kB is needed. Thus, the use of the PDE method can benefit network transmission of medical data for applications of medical visualization, such as telemedicine.

Moreover, the head surface can be reconstructed with different numbers of PDE patches, and the number of PDE patches influences the reconstructed detail. Generally speaking, the larger the number of PDE patches involved, the more geometric detail the surface reconstruction can recover. The PDE reconstruction results with the patch

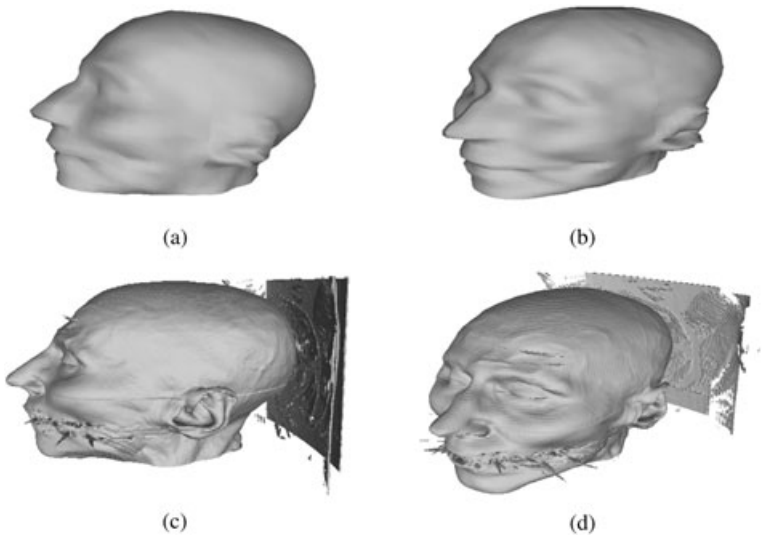


Figure 11. The head surface reconstruction results. (a and b) Results generated by our partial differential equation method using 31 slices from the dataset. (c and d) Reconstruction results produced by the marching cubes algorithm using all the 99 slices of the dataset.

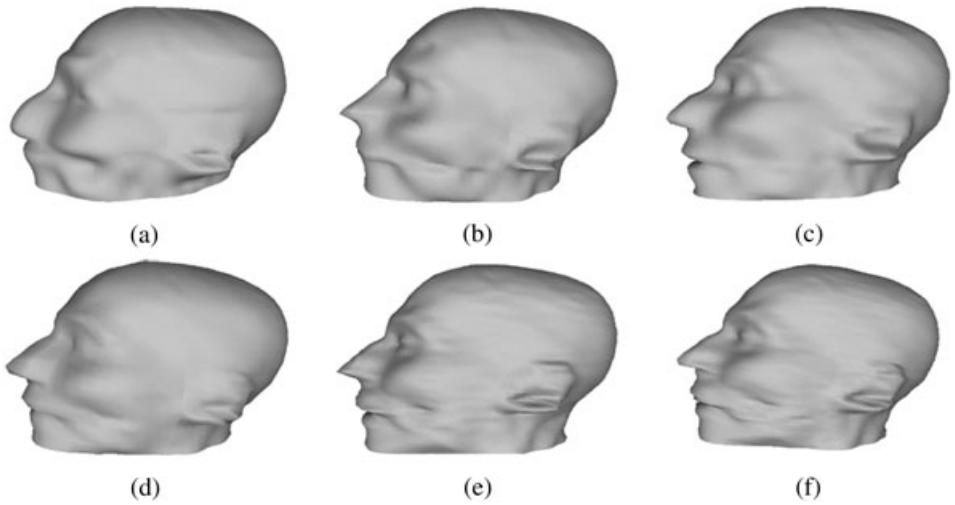


Figure 12. The head surface reconstruction results when (a) $t = 4$, (b) $t = 5$, (c) $t = 6$, (d) $t = 10$, (e) $t = 14$, and (f) $t = 31$.

number gradually increased are displayed in Figure 12. It can be seen that when t reaches 31, that is, 94 out of 99 slices are used, the reconstruction result appears as detailed as the result produced by the MC algorithm shown in Figure 11(c). If t is set to 4, then the facial features are smoothed out. It is also observed that when t is chosen as 10 or 14, the results retain almost all the facial features including eyes, nose, and mouth. This observation reveals one major characteristic of the PDE method we explore here, that is, our PDE method needs only a little amount of information to represent complex geometry. Thus, our PDE method is computationally efficient and fast.

The geometric detail of the reconstructed surface can also be controlled by variant uv resolutions. As shown in Figure 13, the test was carried out with the same PDEs and boundary conditions as in Figure 11(a) by fixing either u or v, while gradually increasing the sampling rate of the other. It can be seen that the geometric detail of the head surface is gradually unveiled.

5.3. Head Volume Rendering Results

Volume rendering is also integrated into the PDE-based framework, and the head solid model has to be generated ahead of volume rendering. Under the PDE-based framework, the head solid model can be generated by introducing a third parameter w into the solution, and the u, v parameters affect the generation of the solid model in the same

DOI: 10.1002/cav

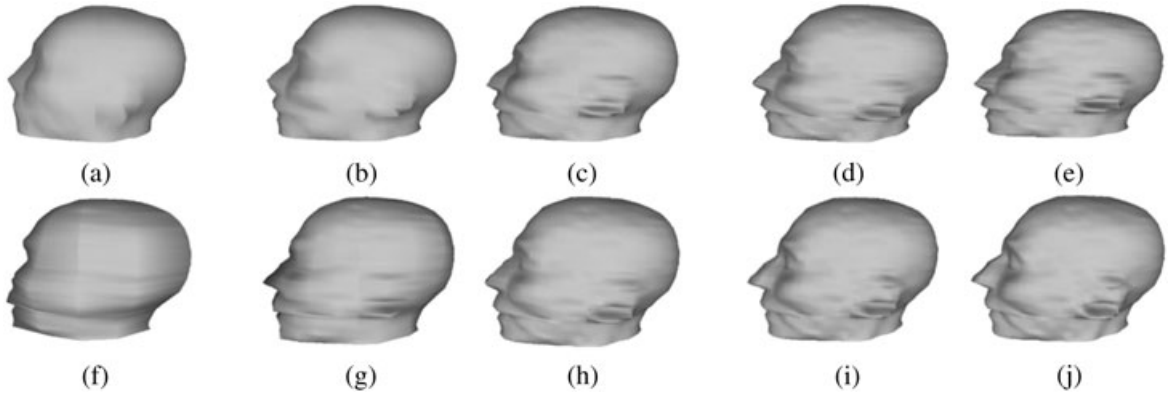


Figure 13. The head surface reconstruction results by adjusting the *uv* resolution. (a) u=10, v=40. (b) u=20, v=40. (c) u=40, v=40. (d) u=60, v=40. (e) u=80, v=40. (f) u=60, v=10. (g) u=60, v=20. (h) u=60, v=40. (i) u=60, v=60. (j) u=60, v=80.

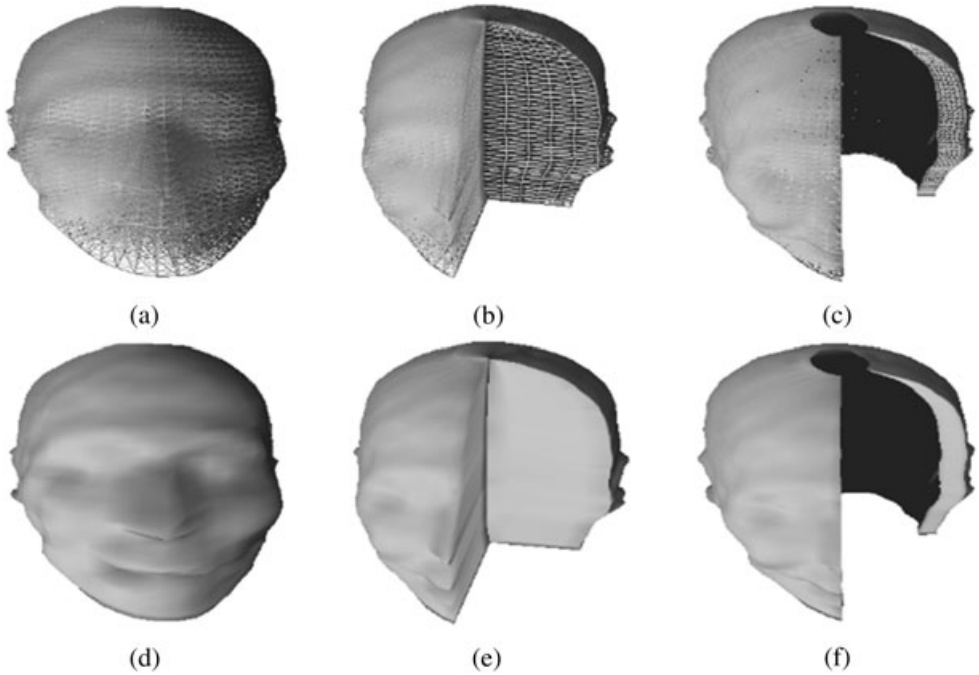


Figure 14. Solid representation of the head. The top row is the grid view, and the bottom row is the smooth view. (a and d) $0 \leq u \leq 1, 0 \leq v \leq 2\pi, 0 \leq w \leq 1$; (b and e) $0 \leq u \leq 1, \pi/2 \leq v \leq 2\pi, 0 \leq w \leq 1$; (c and f) $0 \leq u \leq 1, \pi/2 \leq v \leq 2\pi, 0.8 \leq w \leq 1$.

manner as in surface reconstruction. Besides, the range of *uvw* parametric domain also determines the reconstructed head solid model. Figure 14 shows some solid head models generated by the PDE method in different views, where the solid head is generated by sampling u, v, and w with 40, 60, and 5 equally spaced points, respectively. Note that the boundary conditions used in head solid modeling are the same as in head surface reconstruction.

Following head solid modeling, 3D texture mapping is carried out. This paper implements the 3D texture mapping using the OpenGL library, by mapping the 3D texture into the head solid model to produce volume rendering results. Figure 15(a) and (b) show the results produced by the intuitive texture mapping method. Figure 15(c)–(f) are generated by our PDE method. The alpha value of the generated texture is set in a linear relation with the parameter w in our PDE method and constrained within the range between 0.1 and 0.9; thus, a perspective effect can be achieved. Moreover, in Figure 15(a), there exists a large amount of noise, compared with Figure 15(c) and (e) rendered by our method, where noise is successfully suppressed. To this end, without any transfer function, our PDE method can still produce visually translucent and noiseless rendering results.

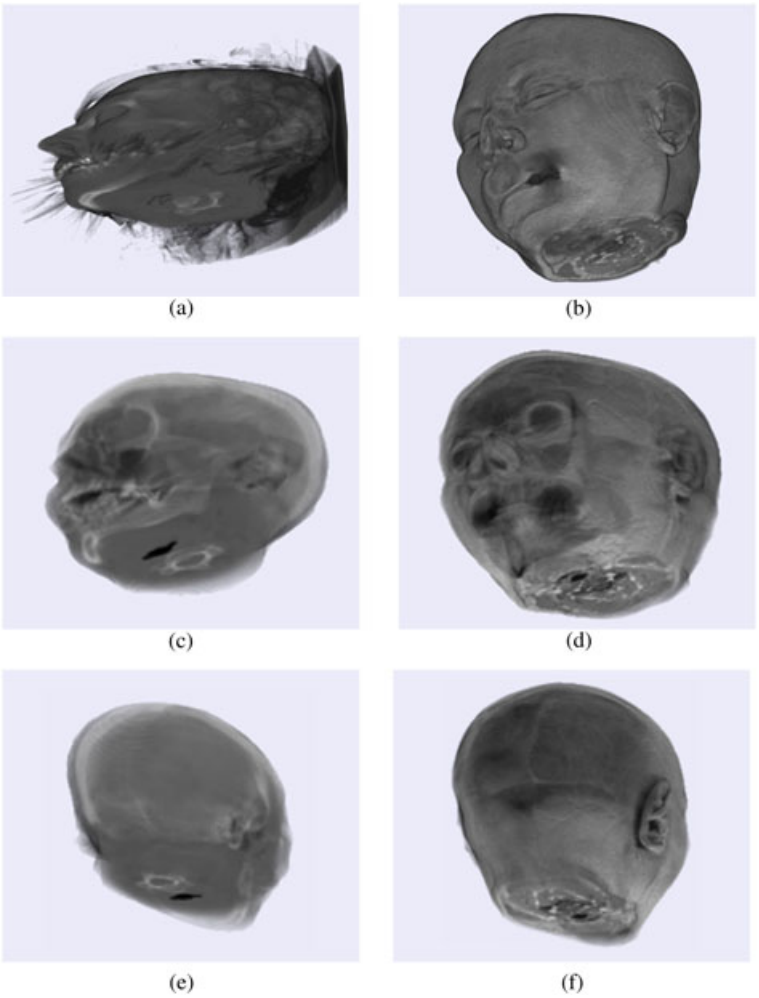


Figure 15. Volume rendering results. (a and b) The result produced by the intuitive texture-based volume rendering algorithm. (c–f) Results produced by our method in different views.

6. CONCLUSION

In this paper, we present a PDE-based head visualization framework, into which both head surface reconstruction and volume rendering are integrated. Our framework first segments a selected number of CT slices, followed by boundary condition extraction. After that, the extracted boundary conditions are input to the PDE engine to perform head visualization. The PDE-based head visualization framework can reconstruct the multiresolutional head surface model using only a small number of head slices and is more efficient in both computational time and storage cost than the conventional MC algorithm. Moreover, without designing a complicated transfer function, our PDE method can render the volume into translucent visualization results in a fast and efficient fashion and suppress the input noise embedded in the medical slices nicely. Note that our volume rendering results are, however, not the best among those existing ones in terms of visual quality but are still visually acceptable with much details reconstructed. The experimental results demonstrate the feasibility of our PDE method for head medical data visualization, including both head surface reconstruction and volume rendering.

Because our method needs only a small number of medical slices, the number of which has to be empirically determined. The selected slices must be able to best represent the geometric detail of the head. Hence, an automatic scheme enabling intelligent selection of the medical slice images for PDE boundary extraction is yet to be sought after.

We believe that the development of the PDE method may benefit many medical visualization applications, such as rendering large-scale medical data with limited resource as well as medical visualization on mobile devices. Rendering large-scale medical data requires high-speed, large-storage capabilities, while mobile 3D medical visualization displays medical data with mobile devices that have limited data storage and transmission abilities. With a small number of boundary conditions, our PDE method can facilitate the representation and storage of medical data. For

DOI: 10.1002/cav

instance, instead of storing a full scale of medical slice images, one may only need to store the boundary conditions of the head geometry, saving a considerable amount of memory.

The focus of this paper is on head medical data visualization. However, our PDE method can be easily applied to any cylinder-like organs. As for these organs with branches, the topology of the connected PDE patches will vary. This will be one of our future tasks based on our work reported in this paper.

ACKNOWLEDGEMENT

This work has been supported by the National Natural Science Foundation of China (61202291).

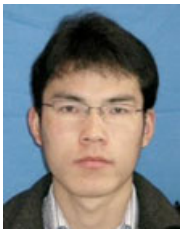
REFERENCES

1. Bloor MIG, Wilson MJ. Generating blend surfaces using partial differential equations. *Computer-Aided Design* 1989; **21**(3): 165–171.
2. Ugail H, Bloor MIG, Wilson MJ. Techniques for interactive design using the PDE method. *ACM Transactions on Graphics (TOG)* 1999; **18**(2): 195–212.
3. Bloor MIG, Wilson MJ. Using partial differential equations to generate free-form surfaces. *Computer-Aided Design* 1990; **22**(4): 202–212.
4. Sheng Y, Sourin A, Castro GG, Ugail H. A PDE method for patchwise approximation of large polygon meshes. *The Visual Computer* 2010; **26**(6-8): 975–984.
5. Ahmat N, Ugail H, Castro GG. Method of modelling the compaction behaviour of cylindrical pharmaceutical tablets. *International Journal of Pharmaceutics* 2011; **405**(1): 113–121.
6. Bloor MIG, Wilson MJ. Efficient parametrization of generic aircraft geometry. *Journal of Aircraft* 1995; **32**(6): 1269–1275.
7. Dekanski CW, Bloor MIG, Wilson MJ. Partial differential equation surface generation and functional shape optimization of a swirl port. *Journal of Propulsion and Power* 1997; **13**(3): 398–403.
8. Castro GG, Ugail H. Shape morphing of complex geometries using partial differential equations. *Journal of Multimedia* 2007; **2**(6): 15–25.
9. Pang M, Sheng Y, Sourin A, Castro GG, Ugail H. Automatic reconstruction and Web visualization of complex PDE shapes. In *2010 International Conference on Cyberworlds (CW)*. IEEE: Singapore, 2010; 97–104.
10. Vandermeulen D, Claes P, Suetens P, De Greef S, Willems G. Volumetric deformable face models for cranio-facial reconstruction. In *Image and Signal Processing and Analysis, 2005. ISPA 2005. Proceedings of the 4th International Symposium on*. IEEE: Zagreb, Croatia, 2005; 353–358.
11. Vanezis P, Vanezis M, McCombe G, Niblett T. Facial reconstruction using 3-D computer graphics. *Forensic Science International* 2000; **108**(2): 81–95.
12. Cesarani F, Martina MC, Grilletto R, Boano R, Roveri AMD, Capussotto V, Giuliano A, Celia M, Gandini G. Facial reconstruction of a wrapped Egyptian mummy using MDCT. *American Journal of Roentgenology* 2004; **183**(3): 755–758.
13. Wilkinson C, Brier B, Neave R, Smith D. The facial reconstruction of Egyptian mummies and comparison with the Fayum portraits: mummies in a new millennium. In *Mummies in a New Millennium: Proceedings of the 4th World Congress on Mummy Studies, Nuuk, Greenland*, 2001; 141–146.
14. Aoki Y, Terajima M, Nakashima A, Hashimoto S. Physics-based 3D head model reconstruction from cephalograms for medical application. *Electronics and Communications in Japan (Part III: Fundamental Electronic Science)* 2002; **85**(11): 45–55.
15. Koch RM, Gross MH, Carls FR, von Büren DF, Fankhauser G, Parish YIH. Simulating facial surgery using finite element models. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*. ACM: New Orleans, USA, 1996; 421–428.
16. Kumar T, Vijai A. 3d reconstruction of face from 2D CT scan images. *Procedia Engineering* 2012; **30**: 970–977.
17. Fuchs H, Kedem ZM, Uselton SP. Optimal surface reconstruction from planar contours. *Communications of the ACM* 1977; **20**(10): 693–702.
18. Lorensen WE, Cline HE. Marching cubes: a high resolution 3D surface construction algorithm. In *ACM SIGGRAPH Computer Graphics*, Vol. 21. ACM: Anaheim, California, 1987; 163–169.
19. Chan SL, Purisima EO. Molecular surface generation using marching tetrahedra. *Journal of Computational Chemistry* 1998; **19**(11): 1268–1277.
20. Cline HE, Lorensen WE, Ludke S, Crawford CR, Teeter BC. Two algorithms for the three-dimensional reconstruction of tomograms. *Medical Physics* 1988; **15**(3): 320–327.
21. Levoy M. Efficient ray tracing of volume data. *ACM Transactions on Graphics (TOG)* 1990; **9**(3): 245–261.
22. Borzecki M, Skurski A, Kamiski M. Applications of ray-casting in medical imaging. *Information Technologies in Biomedicine* 2014; **3**: 3–14.
23. Wallis JW, Miller TR, Lerner CA, Kleerup EC. Three-dimensional display in nuclear medicine. *IEEE Transactions on Medical Imaging* 1989; **8**(4): 297–230.

24. Lacroute P, Levoy M. Fast volume rendering using a shear-warp factorization of the viewing transformation. In *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*. ACM: New York, 1994; 451–458.
25. Van Gelder A, Kim K. Direct volume rendering with shading via three-dimensional textures. In *Volume Visualization, 1996. Proceedings., 1996 Symposium on*. IEEE: San Francisco, 1996; 23–30.
26. Kronander J, Jonsson D, Low J, Ljung P. Efficient visibility encoding for dynamic illumination in direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics* 2012; **18**(3): 447–462.
27. Smelyanskiy M, Holmes D, Chhugani J, Larson A, Carmean DM, Hanson D, Dubey P, Augustine K, Kim D, Kyker A, *et al.* Mapping high-fidelity volume rendering for medical imaging to CPU, GPU and many-core architectures. *IEEE Transactions on Visualization and Computer Graphics* 2009; **15**(6): 1563–1570.
28. Strengert M, Klein T, Botchen R, Stegmaier S, Chen M, Ertl T. Spectral volume rendering using GPU-based raycasting. *The Visual Computer* 2006; **22**(8): 550–561.
29. Rodrłguez B, Gobbetti M, Iglesias Guitin E. State of the art in compressed GPU based direct volume rendering. *Computer Graphics Forum* 2014; **33**(6): 77–100.
30. Chen W, Chen W, Bao H. An efficient direct volume rendering approach for dichromats. *IEEE Transactions on Visualization and Computer Graphics* 2011; **17**(12): 2144–2152.
31. Jonsson D, Sundn E, Ynnerman A, Ropinski T. A survey of volumetric illumination techniques for interactive volume rendering. *Computer Graphics Forum* 2014; **33**(1): 27–51.
32. Kirmizibayrak C, Wakid M, Yim Y, Hristov D, Hahn JK. Interactive focus + context medical data exploration and editing. *Computer Animation and Virtual Worlds* 2014; **25**(2): 129–141.
33. Ferre M, Puig A, Tost D. A framework for fusion methods and rendering techniques of multimodal volume data. *Computer Animation and Virtual Worlds* 2004; **15**(2): 63–77.
34. Noguera JM, Jimnez JR, Ogyar CJ, Segura RJ. Volume rendering strategies on mobile devices. In *The International Conference on Computer Graphics Theory and Applications*, Rome, Italy, 2012; 447–452.
35. Kubiesa S, Ugail H, Wilson M. Interactive design using higher order PDEs. *The Visual Computer* 2004; **20**(10): 682–693.
36. Ugail H. Generalized partial differential equations for interactive design. *International Journal of Shape Modeling* 2007; **13**(02): 201–223.
37. Sheng Y, Willis P, Castro GG, Ugail H. Facial geometry parameterisation based on partial differential equations. *Mathematical and Computer Modelling* 2011; **54**(5): 1536–1548.
38. You L, Zhang JJ, Comninos P. Blending surface generation using a fast and accurate analytical solution of a fourth-order PDE with three shape control parameters. *The Visual Computer* 2004; **20**(2-3): 199–214.
39. Zhang J, You L, *et al.* Fast surface modelling using a 6th order PDE. In *Computer Graphics Forum*. Wiley Online Library, 2004; 311–320.
40. Du H, Qin H. Free-form geometric modeling by integrating parametric and implicit PDEs. *IEEE Transactions on Visualization and Computer Graphics* 2007; **13**(3): 549–561.
41. Du H, Qin H. Integrating physics-based modeling with pde solids for geometric design. In *Computer Graphics and Applications, 2001. Proceedings. Ninth Pacific Conference on*. IEEE, 2001; 198–207.
42. Bloor MIG, Wilson MJ. Spectral approximations to PDE surfaces. *Computer-Aided Design* 1996; **28**(2): 145–152.
43. Aubert G, Kornprobst P. *Mathematical problems in image processing: partial differential equations and the calculus of variations*, Vol. 147. Springer, 2006.
44. Chan TF, Vese LA. Active contours without edges. *Image Processing, IEEE Transactions on* 2001; **10**(2): 266–277.
45. Rudin LI, Osher S, Fatemi E. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena* 1992; **60**(1): 259–268.
46. Chambolle A. An algorithm for total variation minimization and applications. *Journal of Mathematical Imaging and Vision* 2004; **20**(1-2): 89–97.
47. Feb 15, 2014. http://graphics.stanford.edu/data/voldata/voldata.html#bunny.
48. Feb 20, 2014. https://mri.radiology.uiowa.edu/visible_human_datasets.html.
49. Feb 18, 2014. http://www.osirix-viewer.com/datasets.
50. Cárdenes R, Bach M, Chi Y-V, Marras I, de Luis R, Anderson M, Cashman P, Bultelle M. Multimodal evaluation method for medical image segmentation. *Computerized Medical Imaging and Graphics* 2009; **96**(EPFL-ARTICLE-117432): 108–124.

DOI: 10.1002/cav

AUTHORS' BIOGRAPHIES



Congkun Chen is a Graduate student in the Department of Computer Science and Technology from East China Normal University. He has been working on applications of partial differential equations in areas such as 3D medical visualization and geometric modeling. Currently, he is working in a project funded by the NSFC on mesh filtering using partial differential equations.

Yun Sheng received the PhD degree in 2006 from the University of Surrey, UK. He was a research fellow working on the EU FP6 project VISNET in the Center for Communication Systems Research at the University of Surrey, and later as a research officer on the EPSRC project, Efficient Geometry Parameterization for Modeling and Animation in the Department of Computer Science, the University of Bath, UK, before he joined the Nanyang Technological University, Singapore, in 2009, as a research fellow, pursuing his research interest in PDEs for computer graphics applications. Currently, he is an associate professor at the East China Normal University. His research interests include PDE-based parameterization and modeling, 3D medical imaging, NPR, shape analysis, facial synthesis and animation, and NUI.

Fang Li received her MSc degree in Mathematics from the South West China Normal University in 2004 and received her PhD degree in Mathematics from the East China Normal University in 2007. She worked in the Department of Mathematics, East China Normal University since 2007. Currently, she is an associate professor. Her research interests include anisotropic diffusion filtering, the variational methods, and PDEs in image processing.

Guixu Zhang received his PhD degree from the Institute of Modern Physics, Chinese Academy of Sciences, Lanzhou, China, in 1998. He is currently a Professor with the Department of Computer Science and Technology, East China Normal University, Shanghai, China. His current research interests include hyperspectral remote sensing, image processing, and artificial intelligence. He serves as an Associate Editor of the Journal of Applied Remote Sensing.

Hassan Ugail is the director of the Center for Visual Computing at Bradford. He has a first class BSc Honours degree in Mathematics from King's College London and a PhD in the field of Geometric Design from the School of Mathematics at University of Leeds. Prof. Ugail's research interests include geometric modeling, computer animation, and functional design. Prof. Ugail has a number of patents on novel techniques relating to geometry modeling, animation, and 3D data exchange. He is a reviewer for various international journals, conferences and grant-awarding bodies. His recent innovations have led to the formation of a university spin-out company, Tangentix Ltd, with investments from venture capitalists. He is also the recipient of the University of Bradford Vice Chancellor's Award for Knowledge Transfer, which he received for his outstanding contribution to research and knowledge-transfer activities.