Automatic 3D Tooth Segmentation using Convolutional Neural Networks in Harmonic Parameter Space

Jianda Zhang[a,b], Chunpeng Li[b], Qiang Song[b], Lin Gao[b,c], Yu-Kun Lai[d]

[a] *University of Chinese Academy of Sciences, Beijing, China*
[b] *Institute of Computing Technology, Chinese Academy Of Sciences, Beijing, China*
[c] *Shenzhen Research Institute of Big Data, Shenzhen 518172, China*
[d] *School of Computer Science and Informatics, Cardiff University, UK*

Abstract

Automatic segmentation of 3D tooth models into individual teeth is an important step in orthodontic CAD systems. 3D tooth segmentation is a mesh instance segmentation task. Complex geometric features on the surface of 3D tooth models often lead to failure of tooth boundary detection, so it is difficult to achieve automatic and accurate segmentation by traditional mesh segmentation methods. We propose a novel solution to address this problem. We map a 3D tooth model isomorphically to a 2D harmonic parameter space and convert it into an image. This allows us to use a CNN to learn a highly robust image segmentation model to achieve automated and accurate segmentation of 3D tooth models. Finally, we map the image segmentation mask back to the 3D tooth model and refine the segmentation result using an improved Fuzzy-Clustering-and-Cuts algorithm. Our method has been incorporated into an orthodontic CAD system, and performs well in practice.

Keywords: Tooth segmentation, Convolutional Neural Networks, Dental mesh, Maximum flow, Surface parameterization

1. Introduction

In recent years, with the development of computer technology and the improvement of three-dimensional (3D) scanning equipment, computer aided design (CAD) systems appear in increasingly more fields. It uses computers and graphics equipment to help designers efficiently accomplish laborious and repetitive tasks. At present, orthodontic CAD systems play an important role in the field of modern dentistry. It first uses 3D scanning equipment to collect 3D tooth model data as input, and then assists the dentist to process the tooth model, in order to simulate the treatment effect, greatly reducing the dentist burden.

*Corresponding author: cpli@ict.ac.cn (Chunpeng Li)
Email addresses: zhangjianda@ict.ac.cn (Jianda Zhang), songqiang@ict.ac.cn (Qiang Song), gaolin@ict.ac.cn (Lin Gao), LaiY4@cardiff.ac.uk (Yu-Kun Lai)

Tooth segmentation is a key part of an orthodontic CAD system. At present, many methods of tooth segmentation have been proposed, but fully automatic segmentation is still a difficult problem. Although human teeth have some basic geometric characteristics, there are differences between the teeth of different people, especially for the teeth of patients, which often have severe deformity, even tooth decay, tooth loss and other conditions. These conditions lead to the lack of robustness for traditional segmentation methods based on geometric features, so it is difficult to achieve the goal of automatic accurate segmentation by setting some fixed parameters. According to the prior knowledge of tooth geometry, there are obvious negative curvature features at the tooth boundary. The curvature based methods [1, 2, 3, 4, 5, 6] usually detect these negative curvature features and divide the surface into different parts. However, there are also negative curvature characteristics on the surface of teeth and gums, which can form noise and cause serious interference. In addition, for smoother meshes, the negative curvature feature is not obvious enough, which can easily cause wrong segmentation.

In order to improve the robustness, some researchers added human computer interaction mechanism in the segmentation process [7, 8], where the user provides some segmentation prior knowledge or manually repairs wrong segmentation, but this would rely too much on user interaction and significantly increase the user burden.

In recent years, it has become an active research topic to solve problems in the fields of computer graphics and computer vision in a data-driven way, and Convolutional Neural Networks (CNNs) have achieved state-of-the-art performance for segmentation tasks on various open source datasets [9]. However, since the mesh data format is quite different from the input format of typical neural networks, it cannot be directly inputted to standard CNNs for training. For this reason, some researchers have proposed methods [10] to encode meshes into images so as to apply the convolutional neural network to the task of processing mesh-based data. At present, the data-driven 3D tooth model segmentation method [11] can achieve satisfactory automatic segmentation results, but their method requires the use of a large-scale 3D tooth model data set with manual labeling (more than 2,000), which is usually difficult to obtain. In addition, their method encodes every face of the dental mesh into a 20×30 image. Encoding 2,000 dental meshes with 200,000 faces in this way will result in data explosion, leading to overly complex calculation and high information redundancy.

In this paper, we present an algorithm for 3D tooth model segmentation in a harmonic parameter space. Mesh surface parameterization [12] is a process of mapping a mesh surface to a parameter space, in order to construct the isomorphic mapping from the original mesh to the parameter space while minimizing the distortion during mapping. We first map a 3D dental mesh isomorphically to a two-dimensional (2D) harmonic parameter space as a 2D mesh, and then sample the 2D mesh to form an image-like structure (i.e., a matrix). The mesh features such as curvatures are encoded as pixel values to generate a 256×1024 image with small data size and fast computing speed. Then we input the image into an image-based CNN to train a robust image segmentation model. Since large-scale 3D model training sets are difficult to obtain, our approach only requires a small amount of 3D training model data to generate geometry images, which compactly capture geometric characteristics in a regular domain, making it possible to train the CNN with limited 3D tooth data. We evaluate

the effectiveness of our method using cross validation.

In the test stage, the test image is inputted into the segmentation model to obtain the image segmentation mask. We then map the image segmentation mask back to the 2D mesh to obtain the label of each vertex of the dental mesh. Since the result of image segmentation can have some deviation at the boundary, and the projection and backprojection between the 2D mesh and the image are not bijective (due to multiple vertices mapped to the same pixel), the boundary of mesh segmentation is not accurate enough. Therefore, we adapt and improve the fuzzy-clustering-and-cuts (FCC) method [13] to address this problem. The FCC method is a mesh segmentation method based on a network flow algorithm [14]. It can detect the path with the minimum concave dihedral angle in a given fuzzy region, and therefore divides the fuzzy region into two parts with the segmentation boundary well aligned with geometric features. We improve the FCC method to make the segmentation boundary more accurate and smooth, taking into account significantly denser mesh triangulation in the boundary regions.

The tooth segmentation method in harmonic parameter space proposed in this paper can automatically and accurately segment different kinds of teeth. With the help of a professional dentist, we manually label 100 dental meshes. The average segmentation accuracy of our method reaches 98.87%, and the Directional Cut Estimate (DCD) is $0.046mm$, which is comparable or better than state-of-the-art methods, including recent deep learning based method [11] that requires much larger training set. Our technique is also applied in an orthodontic CAD system, and achieves good performance in practice.

1.1. Contributions

The main contributions of this paper are as follows:

- We present a novel tooth model segmentation method in harmonic parameter space, which can achieve automatic and accurate segmentation and is well applied in an orthodontic CAD system;
- We design an image segmentation scheme of CNN, and generate a unique dataset through data enhancement for CNN training and testing. The trained model has strong robustness and good generalization performance for new tooth geometric images.
- We improve the fuzzy-clustering-and-cuts (FCC) method, which can detect the concave segmentation boundary more accurately and make the segmentation boundary smoother.

The rest of this paper is organized as follows: We first summarize related work in the field, and then describe an overview over the segmentation process and give the method details. We then present experimental results and analysis, and finally we draw conclusions of the paper.

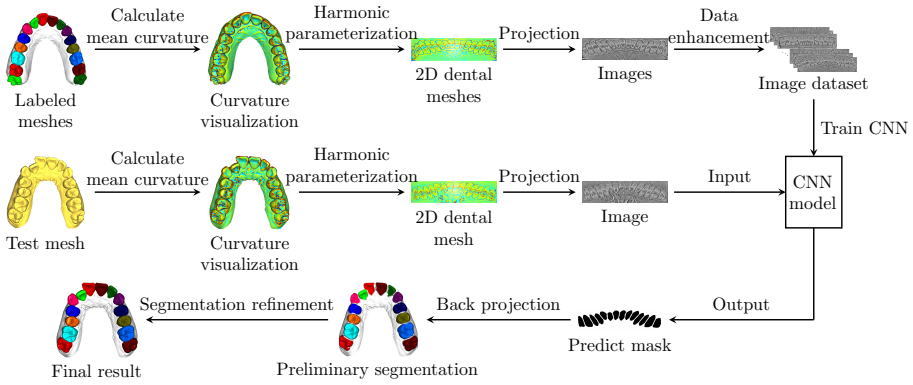


Figure 1: The pipeline of our approach. We input the original tooth mesh, and after a series of processing, we finally get the segmentation label of each vertex of the tooth mesh. The top line shows the process of obtaining the image training set by harmonic parameterization, projection, data enhancement and other operations of the labeled tooth meshes. The middle line shows the process of obtaining the tooth image from the mesh to be segmented through similar operations and input into the CNN model. The bottom line shows the image segmentation mask output by the CNN network, backprojected to the mesh to be segmented, and then refined.

2. Related Work

This paper focuses on dental mesh segmentation, which is an important application of mesh segmentation. We introduce general mesh segmentation methods, followed by dental specific methods.

2.1. General Mesh Segmentation

3D mesh segmentation is a key part of computer graphics [15]. It divides a mesh into different parts according to some reasonable rules. Common methods can be divided into two categories: region-based methods and boundary-based methods. Region-based methods gather similar regions together according to the geometric information of the mesh. Well-known region-based works include K-means [16, 17], clustering [18], hierarchical decomposition [13], primitive fitting [19], watersheds [20], random walks [21]. Boundary-based methods instead detect the geometric feature boundaries of the mesh which divide the mesh into different parts. Such methods include randomized cuts [22], fuzzy-clustering-and-cuts (FCC) [13], core extraction [23], shape diameter function [24], active contours or scissoring [25, 26], and sparse and low-rank representation [27]. However, these methods rely too much on the geometric information of the mesh, and often fail once the mesh becomes too complex.

As meshes can have significant variation, it is challenging to separate a mesh into desired parts with a fully automatic approach. Manual segmentation on the other hand is labor intensive and time consuming. So sketch-based semi-automatic methods become popular. They provide simple and user-friendly interfaces for users to add their suggestions as starting points or optimization constraints. For example, Ji et al. [28] introduced an improved region-growing algorithm for segmentation. Fan et al. [29]

adopted an efficient local graph-cut based optimization algorithm and received satisfactory results. Studies [30, 31, 32, 33] integrated harmonic field theory with sketch-based segmentation, which possess solid theoretical basis and work well. Khan et al. [34] proposed to use a robust interactive segmentation method to improve remeshing quality. In their approach, mesh segmentation is first initialized using an existing interactive method based on the live-wire interaction [35] to well capture sharp features. This is then refined using local operations on vertices and edges to improve the segmentation. However, sketch-based methods need to reach a balance between user input and automatic computation.

Since 3D mesh databases such as the Princeton Segmentation Benchmark [36] were released, data-driven methods have been proposed for mesh segmentation. Both supervised and semi-supervised learning methods aim to learn a model for segmenting meshes into meaningful parts, using a labeled training set; some recent works include [37, 38, 39, 40].

2.2. Dental Mesh Segmentation

Numerous segmentation approaches have been proposed to separate tooth models. According to the input format, we divide the existing approaches into two categories: 3D mesh-based methods and 2D image-based methods.

3D mesh based methods can be further divided into two types, namely curvature-based methods, and harmonic-field-based methods. Curvature-based methods are the majority. Yuan et al. [3] analyzed the regions of the 3D tooth model and classified them based on the minimum curvatures of the surface. Zhao et al. [41] proposed an interactive segmentation method based on curvature values of the triangle mesh. The system designed by [4] requires users to provide a one-time setting of a certain curvature threshold via an intuitive slider. Others, including snake-based active contour method [5], fast marching watershed method [6] and morphological skeleton extraction method [1] are all related to curvature information to some extent.

Harmonic-field-based methods are in the minority. Zou et al. [42], Liao et al. [43] and Li et al. [44] applied harmonic fields to tooth segmentation, which require only a limited number of surface points as prior. It saves users time and achieves reasonable results.

Researchers have also proposed effective segmentation algorithms based on the 2D projection images. Yamany and El-Bialy [2] encoded the curvature and surface normal information into a 2D image, and designed an image segmentation tool to extract structures of high/low curvatures. Similarly, Toshiaki et al. [45] presented an automated method for tooth segmentation from 3D digitized images captured by a laser scanner. Grzegorzek et al. [46] presented a multi-stage approach for tooth segmentation from 3D dentition surfaces based on a 2D model-based contour retrieval algorithm. Wongwaen et al. [47] converted the 3D-panoramic to 2D space to find the cutting points for segmentation of individual teeth, followed by converting the 2D image back to 3D space for remaining operations. Xu et al. [11] used a similar set of features as in [10], to encode every face of a mesh into a 20×30 image. Then they inputted these images into a CNN to train a segmentation model, and finally used the mesh segmentation refinement algorithm to refine the segmentation boundary, achieving satisfactory results. The method however requires a large labeled mesh dataset for training.

3. Method

Our method takes a dental mesh as input and aims to get the label for each vertex of the dental mesh. Figure 1 illustrates our pipeline. The whole method can be divided into three steps, mesh parameterization, image segmentation, and segmentation refinement.

3.1. Mesh Parameterization

The purpose of mesh parameterization is to find a one-to-one mapping of points on a 3D mesh to a parameter space, and to minimize the distortion of a certain geometric metric while maintaining the topological information on the parameter space isomorphic to the original mesh. Our dental mesh is a non-closed genus-zero 3D surface with only one boundary. Geometrically, it is isomorphic to the planar disk topology. Suppose the parameter formulation of surface $M \subset \mathbb{R}^3$ relative to the point (u, v) in a plane space $D \subset \mathbb{R}^2$ is:

$$r(u, v) = (x(u, v), y(u, v), z(u, v)) \tag{1}$$

Convex representation method [48] is a kind of common surface parameterization method, which fixes a polygon boundary ∂D on plane D, and linearly maps the boundary ∂M of surface M to ∂D. Then the coordinates (u, v) of the internal vertices of the plane D can be determined by energy minimization. Our dental models are obtained by scanners and then manually processed to remove all but the gums and the teeth. Therefore, the overall shape of the obtained models in space is similar to an arch, and the boundaries of all models are similar, as shown in Figure 2.

The purpose of surface parameterization is to obtain the geometric image of dental models, so we use a rectangle as the boundary of plane D, which can minimize image redundancy. In addition, considering the characteristics and overall shape of dental models, we set the aspect ratio of the rectangle to $4 : 1$. To map the boundary of the original mesh ∂M to the rectangular boundary ∂D, we first calculate the two vertices (v_i^*, v_j^*) with the largest geodesic distance on the original mesh boundary:

$$(v_i^*, v_j^*) = \arg \max_{(v_i, v_j)} (Dis\_geo(v_i, v_j)), v_i, v_j \in \partial M \tag{2}$$

where $Dis\_geo(v_i, v_j)$ is the geodesic distance between v_i and v_j. Then, we fix (v_i^*, v_j^*) to the midpoints of the two short sides of the rectangle as (h_i^*, h_j^*), and map the remaining vertices on the original boundary to the rectangular boundary. Figure 2 shows the process of mapping the 3D surface boundary ∂M to the plane boundary ∂D.

Using the method of energy minimization to determine the coordinates of the internal vertices of the plane domain D, it is only necessary to solve a linear equation system, which is efficient, and the key lies in the selection of energy weights. [12] presents a mesh parameterization method based on harmonic mapping. Its energy function settings are as follows:

$$E_{harm}(h) = \frac{1}{2} \sum_{e_{i,j} \in M} \kappa_{i,j} \|h_i - h_j\|^2 \tag{3}$$

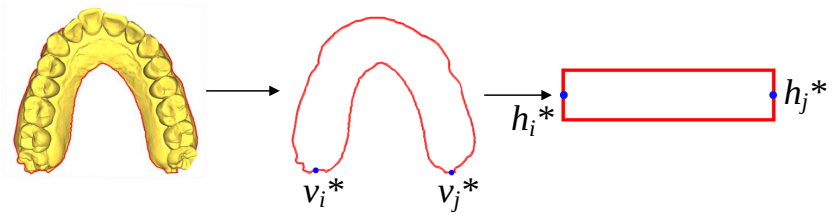


Figure 2: We first detect the boundary of 3D surface, obtain the vertices and edges on the boundary, and then calculate the two vertices (v_i^*, v_j^*) on the boundary with the largest geodesic distance. We fix (v_i^*, v_j^*) to the midpoints of the two short sides of the rectangle as (h_i^*, h_j^*), and map the remaining vertices to the rectangular boundary in proportion to the length of the edges.

where h_i is the vertex on the plane D corresponding to the vertex v_i on the original mesh M, $e_{i,j}$ is the edge of the vertex with v_i and v_j in M, and the spring constants $\kappa_{i,j}$ are computed as follows: For each edge $e_{i,j}$, let $L_{i,j}$ denote its length as measured in the original mesh M, and for each face $f_{i,j,k}$, let $Area_{i,j,k}$ denote its area, again as measured in M. Each interior edge $e_{i,j}$ is incident to two faces, namely f_{i,j,k_1} and f_{i,j,k_2}. Then

$$\begin{aligned}\kappa_{i,j} =&(L_{i,k_1}^2 + L_{j,k_1}^2 - L_{i,j}^2)/Area_{i,j,k_1} + \\ &(L_{i,k_2}^2 + L_{j,k_2}^2 - L_{i,j}^2)/Area_{i,j,k_2}\end{aligned} \tag{4}$$

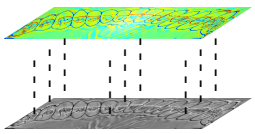
To minimize the energy E_{harm}, we solve the following:

$$\frac{\partial E_{harm}}{\partial h_i} = \sum_{e_{i,j} \in M} \kappa_{i,j}(h_i - h_j), h_i \in (D - \partial D) \tag{5}$$

Solving this sparse linear system of equations gives the coordinates of each internal vertex of the plane D.

The geometry between adjacent teeth is quite complex, the vertices of this area are very dense, and the area of each triangle is small. The harmonic parameterization used guarantees one-to-one mapping. However, multiple triangles may be mapped to the same pixel once discretized to a geometric image. This tends to only affect a small number of (typically 1 or 2) pixels. Moreover, our method has a final segmentation and refinement step. This step is processed on the *original* model to eliminate the effect of such overlapping on the final segmentation result.

The data format of the planar mesh and the image is very different. We project each face on the planar mesh onto the pixel at the corresponding position of the image, and encode the curvature as a pixel intensity value to generate an image with an aspect ratio of 4:1. We then calculate the mean curvature of the discrete mesh, and map the curvature to $[0, 255]$ according to Eq. 6:



$$Cur^{'}(i) = \frac{255 \cdot (\tanh(Cur(i)) + 1)}{2} \tag{6}$$

where $Cur(i)$ is the mean curvature of the vertex v_i. Since mesh data has sub-pixel level accuracy, increasing image resolution preserves more detail. In theory, when the image resolution is high enough, it is guaranteed that the information of each face is preserved, but the image dimension cannot be too high in practice. The number of tooth mesh faces is approximately 200,000, so we project a planar tooth mesh to a 256×1024 image, which achieves a good balance of efficiency and accuracy.

3.2. Image Segmentation

The purpose of image segmentation is to segment the image at the pixel level to get the segmentation mask for each tooth. Mapping a 3D dental mesh to the harmonic parameter space can effectively avoid overlapping of vertices and faces on the planar dental mesh, as the mapping is guaranteed to be isomorphic. Each tooth in the image is independent of each other, so a complete segmentation mask for each tooth can be obtained.

Obtaining the segmentation mask of each tooth from the input geometric image is actually an image entity segmentation task. Unlike semantic segmentation, entity segmentation needs to distinguish multiple entities within one class. However, the characteristics of these entities are very similar, and there is almost no distinction between them. Our geometric images of teeth encode the curvature features of the original mesh into pixels. The adjacent teeth are extremely similar, but the interface of each tooth and other teeth or gums has obvious and relatively complete negative curvature features.

[9] presents an effective medical image segmentation network: U-Net. It takes the original image as input and outputs the segmentation map. We refer to the structure of U-Net and design a dental image segmentation network model. The loss function used is the cross entropy loss.

The segmentation mask of adjacent teeth is prone to being falsely connected, causing segmentation to fail, so we made the following two improvements: The first is to reduce the segmentation mask range of each tooth so that the boundary of the segmentation mask is inside the ground truth tooth boundary, which is equivalent to enlarging the border between adjacent teeth and enhancing the independence between each tooth. The second is to increase the training weight at the tooth boundary, so our loss function is as follows:

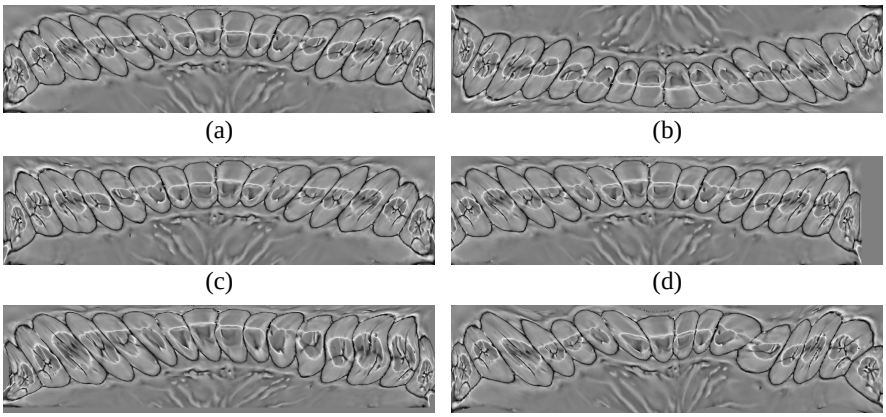
$$\begin{aligned} Loss = -\Bigg(& \rho \sum_{p_i \in B} [p_i \log \hat{p}_i + (1 - p_i) \log(1 - \hat{p}_i)] \\ & + \sum_{p_i \in (I - B)} [p_i \log \hat{p}_i + (1 - p_i) \log(1 - \hat{p}_i)] \Bigg) \end{aligned} \tag{7}$$

where p_i is the predicted value, $\hat{p}_i$ is the ground true value, I is the set of pixels for the entire image, B is the set of boundary pixels in the image, and ρ is the boundary weight. Our statistics show that the average proportion of the boundary area is about 5%, so we set ρ to 20 to balance the two terms.

The image segmentation mask and boundary weight map are shown in Figure 3.

(a) (b) (c)

Figure 3: (a) is the original tooth image. (b) is the image segmentation mask. We set the tooth part to 0 (black) and the other part to 1 (white). (c) is the boundary weight map. We set the black boundary to 10, and the other part to 1.



(a) (b) (c) (d) (e) (f)

Figure 4: (a) is the original image, (b) is the image after the rotation of $180°$, (c) is the image after the horizontal inversion, (d) is the image after the translation of $v(0, -50)$, (e) is the image after the horizontal disturbance, (f) is the image after the vertical disturbance.

Training a network model with good performance requires a large amount of training data, but large-scale 3D tooth model data sets are difficult to obtain, so we use a small number of 3D tooth models to generate geometric images, and then enhance the geometric images through data augmentation for robust training. The details are as follows:

- Rotation: we set the range of rotation angle $\alpha \in [-10°, 10°] \cup [170°, 190°]$, randomly get a rotation angle α_0 from the range according to a uniform distribution, and then rotate the image around the center of the image.
- Flip: we randomly flip each image vertically or horizontally.
- Translation: we set the range of the translation vector $v(dx, dy)$ to be $dx \in [-20, 20], dy \in [-100, 100]$. Then, we randomly obtain a translation vector from the range according to a uniform distribution, and perform translation for each image.
- Sinusoidal disturbance: we add sinusoidal perturbation to the image respectively in the transverse and longitudinal direction, and the image coordinates are con-

verted according to the following:

$$\begin{cases} x_t = x + a\sin(\pi T y) \\ y_t = y, \end{cases} \tag{8}$$

where x and y represent the original pixel coordinates, x_t and y_t represent the pixel coordinates after disturbance, $a \in [10, 15]$ represents the disturbance amplitude and $T \in [0.005, 0.01]$ represents the phase. Eq. 8 represents the horizontal perturbation operation on the image. Similarly, swapping x and y in the equation leads to the longitudinal perturbation operation on the image.

All the parameter settings in the above image augmentation take into account the characteristics of the image. Under the condition that the image is still plausible without significant distortion, the diversity of sample data is increased as much as possible. The upper and lower limits of the parameters are set to ensure that the obtained images are plausible and not visually distorted. Overall, we enlarge the data set by about 40 times through these augmentation operations. Figure 4 shows the comparison before and after data augmentation.

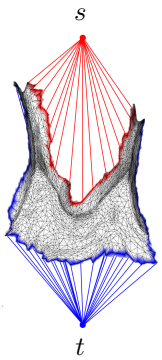
Different dental mesh reconstruction may create tooth models with substantially different accuracy, which will cause the estimated curvature values of these meshes to be different, and the contrast of the corresponding images can vary significantly. Therefore, we apply global contrast normalization [49] for each image to eliminate segmentation errors caused by contrast differences.

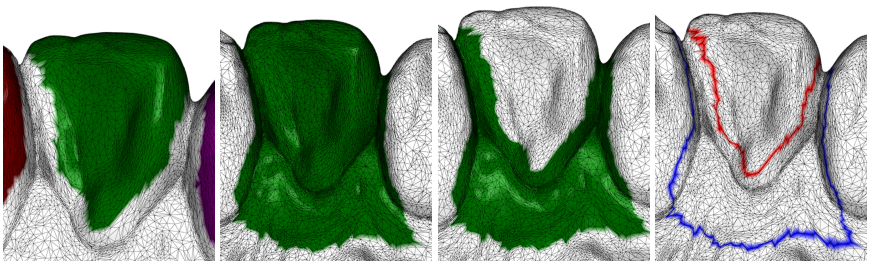
3.3. Segmentation Refinement

After the image is segmented, we get the segmentation mask. The segmentation mask is backprojected to the original mesh to obtain the surfaces $M_i, i = 1, 2, \ldots, n$ of n teeth, and the preliminary segmentation is completed. Assuming that the ground truth surface of each tooth is $\hat{M}_i$, the purpose of segmentation refinement is to find the ground truth segmentation boundary $\partial\hat{M}_i$ on the result of the preliminary segmentation. As shown in Figure 5, in the preliminary segmentation result, each tooth surface boundary ∂M_i is generally inside the ground truth tooth boundary $\partial\hat{M}_i$, so we extend the surface M_i outward to form a surface $M_i^{'}$ which is expected to contain the ground truth tooth boundary. Then the fuzzy region is:

$$M_{fi} = (M_i^{'} - M_i) \cup \partial M_i. \tag{9}$$

[13] presents a mesh segmentation method FCC based on a maximum-flow algorithm [50], which can find the segmentation boundary with the smallest concave dihedral angle in a given fuzzy region, and divide the fuzzy region into two parts.The method first constructs an undirected graph $G =< V, E >$, where V is the set of vertices in M_{fi} and E is the set of edges in M_{fi}. In addition, two virtual nodes s and t are added to the set V to represent the source point and the sink point respectively. The edges are added to E to connect s to each vertex on the boundary ∂M_i, and t to each vertex on the boundary $\partial M_i^{'}$.





(a) (b) (c) (d)

Figure 5: Construction of an undirected graph G for one tooth. In (a), the green region is the surface M_i of a tooth in the preliminary segmentation result; in (b), the green region is the surface $M_i^{'}$ of the tooth after the expansion; In (c), the green region is the surface M_{fi} of the fuzzy region; (d) shows the two boundaries of the fuzzy region M_{fi}, which are also the boundaries of M_i and $M_i^{'}$, where the red boundary is ∂M_i and the blue boundary is $\partial M_i^{'}$.

Using the maximum-flow algorithm to segment the mesh, the most important thing is how to set the capacity $Cap(i,j)$ of each edge. Usually, the two objects in contact with each other have concave dihedral and negative curvature features at the joint surface. By detecting these features, the most reasonable segmentation boundary can be found. [13] uses the concave dihedral feature to set the capacity of each edge according to Eq. 10.

$$Cap(i,j) = \begin{cases} \dfrac{1}{1 + \frac{Ang\_dist(\alpha_{i,j})}{avg(Ang\_dist)}}, & \text{if}(i,j \neq s,t) \\ +\infty, & \text{otherwise}, \end{cases} \tag{10}$$

where $\alpha_{i,j}$ represents the dihedral angle of the edge $e_{i,j}$, $Ang\_dist(\alpha_{i,j})$ is as follows:

$$Ang\_dist(\alpha_{i,j}) = \eta \cos(1 - \alpha_{i,j}) \tag{11}$$

η is a coefficient between 0 and 1. A small positive value (usually 0.1) is used for convex angles and $\eta = 1$ is used for concave angles as concave edges are more important for segmentation.

It has been found through experiments that the result is not ideal. The segmentation boundary is rough, and even deviates significantly from the ground truth tooth boundary $\partial \hat{M}_i$. This is because triangular meshes use many triangular patches to approximate 3D object surfaces, and the vertices and edges in regions with significant negative curvature characteristics tend to be much denser than other flat regions. Although the weight of each edge of this part of the region is small, the path weighting may be large due to the accumulation of a large number of edges, so the path with the smallest weight may deviate from the ground truth tooth boundary $\partial \hat{M}_i$. Therefore, we present an improvement to set the capacity of each edge according to Eq. 12 where the

Items	CPU	GPU	RAM
CNN model	Core i5, 3.3GHz	Nvidia 1080Ti	16G
Others	Core i7, 1.8GHz	None	16G

Table 1: Details of hardware parameters

edge length is also taken into account:

$$Cap_{our}(i,j) = \begin{cases} Cur(i,j) \cdot \|l_{i,j}\|^2, \text{if}(i,j \neq s,t) \\ +\infty \qquad\qquad , \text{otherwise}, \end{cases} \tag{12}$$

$l_{i,j}$ is the length of the edge $e_{i,j}$. $Cur(i,j)$ is defined as follows:

$$Cur(i,j) = \frac{N_e(Cur^{'}(i) + Cur^{'}(j))}{\sum_{e_{i,j} \in M_{fi}}[Cur^{'}(i) + Cur^{'}(j)]} \tag{13}$$

where $Cur^{'}(i)$ is the same as in Eq. 6, N_e is the number of edges in M_{fi}. It can be seen from Eq. 12 that the capacity $Cap_{our}(i,j)$ is the product of two terms: curvature term $Cur(i,j)$ and edge-length term $\|l_{i,j}\|^2$. The curvature term is used to detect negative curvature features, and the edge-length term limits the shortest path to dense regions of vertices and edges, eliminating the negative effects caused by the large number of dense region edges. After the above improvements, we have obtained a more precise and smooth segmentation boundary.

4. Experimental Results and Analysis

To verify the effectiveness of our method, we produced a data set containing 100 tooth models and complete manual labeling with the help of professional dentists. These models come from several different commercial 3D scanners designed to test the universality of our approach. All our experiments were carried out on two different computers, one for training and testing the CNN model and the other for other experiments unrelated to the CNN model. Details of hardware parameters are shown in Table 1.

Our experiments are divided into two parts. The first part is the image segmentation experiment to evaluate the performance of the image segmentation network. The second part is the mesh segmentation refinement experiment, to verify the final segmentation result accuracy.

4.1. Image Segmentation

Our image segmentation network is based on the U-net network structure, inputting 256×1024 single channel images and outputting 256×1024 segmentation masks. Currently, there are no publicly available large-scale dental mesh data sets, and we only have 120 dental models with geometry images. The data of 120 dental models was obtained using two different scanners, and the number of triangles of dental models obtained by the two scanners ranges from 30,000 to 90,000, and from 200,000 to 500,000,

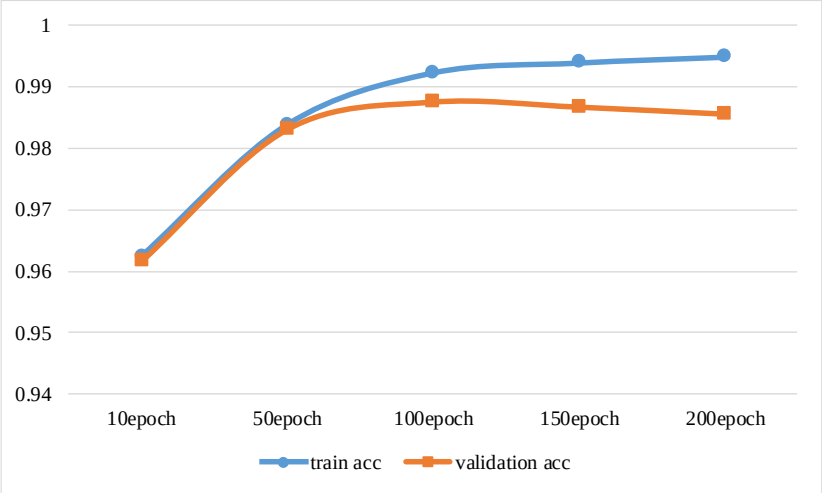
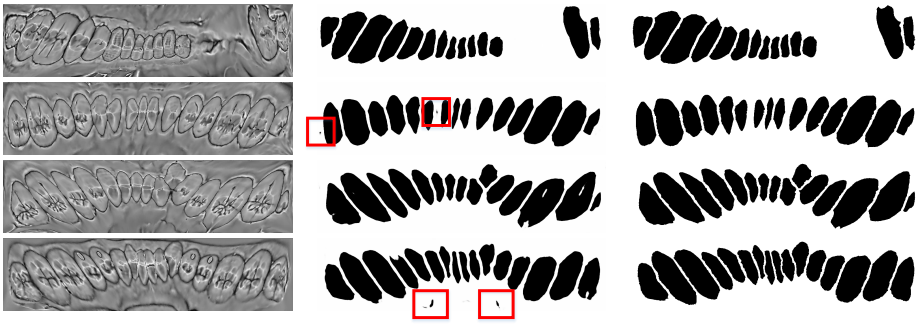


Figure 6: This figure shows the prediction accuracy of models with different training epochs on the training and validation sets. The blue curve is training set, the orange curve is validation set, the vertical axis illustrates the average prediction accuracy of the 5fold cross-validation experiment, and the horizontal axis illustrates the number of training epochs.

respectively. The ratio of the numbers of these two kinds of models is about 4:6. We use stratified random sampling to select 20 models as the test set, and the remaining 100 models as the training set and validation set (with 4:1 split). Training the network with such a small data set may cause over-fitting and make it difficult to verify the network performance. To solve the above problems, we design a 5-fold cross validation experiment. We also use stratified random sampling to sample the remaining data set five times, without duplicated samples. In this way, the data set is evenly divided into five groups, each of which includes 20 models. Using 5-fold cross validation, one group is selected as the validation set and the other four groups are selected as the training set. Each dental model corresponds to a geometric image. Before training the network model, we first enhance the training set as described in Section 3.2. Each group is expanded from the original 20 images to 800 images through data augmentation.

We train the CNN model with different numbers of epochs. Figure 6 shows the prediction accuracy of models with different training epochs on the training and validation sets. As the number of training epochs increases, the prediction accuracy of the training set continues to increase, but the prediction accuracy of the validation set shows a peak near 100 epochs, which means that over-fitting occurs after 100 epochs of model training. Therefore, we choose 100 training epochs, use the entire training set and validation set as the training set, retrain the model, and then use the test set for testing. The average prediction accuracy of the test set is 98.69%.

[11] transforms the dental mesh into matrix format data adapted to CNN input. Through feature extraction, they extract 600-D features from each face of the mesh and generate 20×30 images. For a tooth model, the number of faces usually ranges from tens of thousands to hundreds of thousands, so their method will lead to excessive fea-



(a) (b) (c)

Figure 7: (a) Original images. (b) Prediction masks. (c) Ground truth.

ture dimensions and complicated calculation. In contrast, our method maps each model to an image with fixed size 256×1024, which greatly reduces the feature dimension. Our segmentation results also show the effectiveness of this method. In addition, because the input, output and evaluation methods of our network model are all different from those of theirs, it is meaningless to compare the prediction accuracy of these two network models. Instead, it is more meaningful to compare the accuracy of the final mesh segmentation results, which we will later show.

Figure 7 shows part of the original images, predicted masks and ground truth. The red box highlights inaccurate parts of the prediction masks, which will affect the subsequent segmentation. We detect the area of each black region and treat the region with the area less than a threshold as noise. We calculated the average area of the noise and the average area of the tooth, and found that the latter is generally more than 15 times larger than the former. In fact, each dental mesh has only 16 teeth at most. We calculated the area of each black region, and calculated the mean value of the largest 16 areas, and set the threshold value as one tenth of the mean value. This denoising process works well in our experiments.

4.2. Segmentation refinement

During the cross-validation, for each test example, segmentation of the tooth model in the geometric image domain is obtained. The preliminary segmentation result can then be obtained by projecting the image segmentation mask back to the dental mesh. The preliminary result can be further improved through segmentation refinement, which is the last step of our pipeline. We improve the FCC algorithm and compare the segmentation results before and after the improvement through experiments. Figure 8 shows the segmentation results of some models. The red boxes show complex areas, and our method still performs well. We quantify the results using the following two measures. One is to calculate the percentage of the area of correctly labeled faces [10], which is expressed as:

$$Accuracy = \frac{\sum_{f_{i,j,k} \in M} Area_{i,j,k} g(l_{i,j,k})}{\sum_{f_{i,j,k} \in M} Area_{i,j,k}} \tag{14}$$

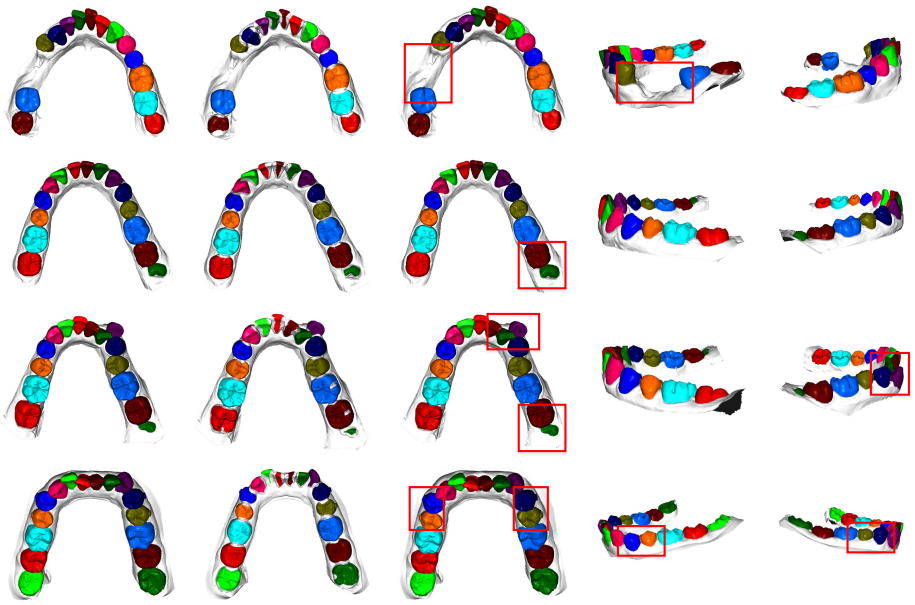


Figure 8: (a) Ground truth. (b) Preliminary segmentation. (c) Final result. (d)&(e) Other views of the final result.

where $Area_{i,j,k}$ is the same as in Eq. 4, $l_{i,j,k}$ is the prediction label of face $f_{i,j,k}$. $g(l_{i,j,k})$ is 1 if the prediction is correct, otherwise 0. Since our outputs are labels of vertices of the mesh, we convert vertex labels to face labels. The three vertices (v_i, v_j, v_k) of face $f_{i,j,k}$ in the mesh have labels (l_i, l_j, l_k). If two or more labels are the same in these three vertex labels, the label $l_{i,j,k}$ of $f_{i,j,k}$ is assigned the vertex label with a majority of the number. The mean segmentation accuracy of our 20 dental meshes reached 98.87%. Another measure is to use Directional Cut Discrepancy (DCD) [36] to calculate the mean error of the segmentation boundary. The DCD of most models is less than 0.1 mm, and the mean DCD of all models is 0.0458 mm.

Compared with the original FCC, our improved segmentation refinement improves the mean segmentation accuracy of all models from 88.2% to 98.87%, and the mean DCD from $0.6127mm$ to $0.0458mm$. Figure 9& Figure 10show the segmentation accuracy comparison results and the DCD comparison results, when comparing our improved refinement with the original FCC, and Figure 11 shows a visual comparison of local details.

Table 2 shows how our approach compares with the latest relevant work. It can be seen that our method achieves comparable accuracy, and much better DCD, compared to state-of-the-art methods. Our method also requires much less training data, compared with existing deep learning method [11]. Note that the performance of [11] was achieved using a much larger training set which is not publicly available, and the performance was reported in their paper.

These models come from different commercial 3D scanners, and the numbers of

Items	Accuracy	DCD
[43]	—	$0.1mm$
[11]	**98.93%**	$0.083mm$
Ours	98.87%	**0.046mm**

Table 2: Comparison of our method with alternative methods for tooth segmentation accuracy and DCD. Our method achieves comparable segmentation accuracy as the state of the art, and significantly lower DCD, demonstrating more accurate segmentation boundaries. Our method only requires a small training set, compared with the existing deep learning method.

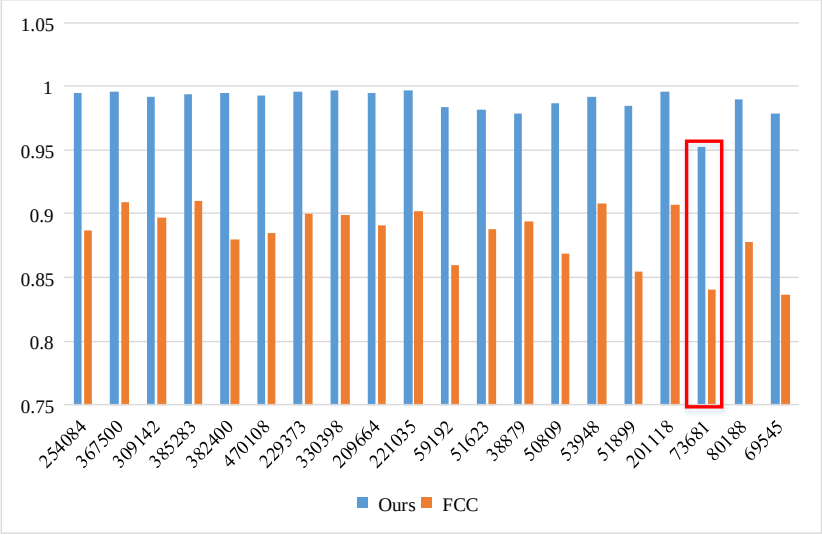


Figure 9: Comparison of our segmentation refinement method and FCC using segmentation accuracy. The blue bars are our improved method, and the orange bars are the FCC method. The horizontal axis illustrates the number of faces of different tooth models. The vertical axis illustrates the segmentation accuracy.

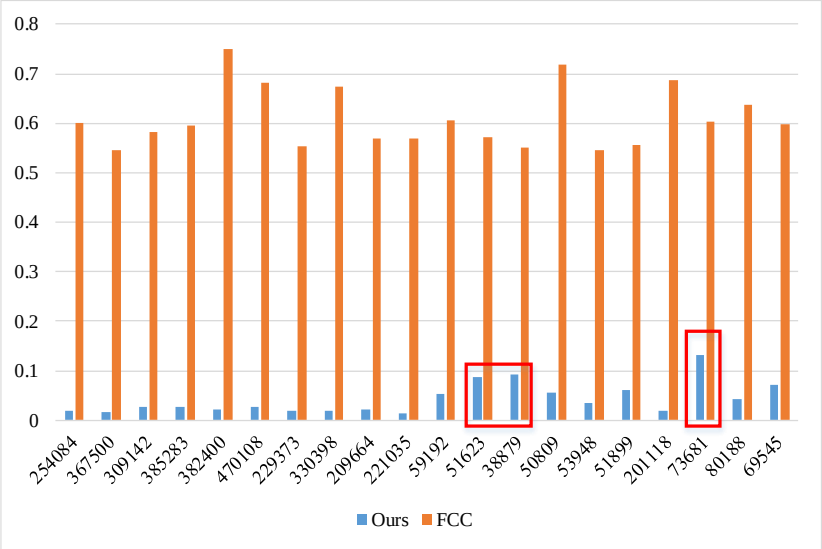


Figure 10: Comparison of our segmentation refinement method and and FCC using DCD measures. The blue bars are our improved method, and the orange bars are the FCC method. The horizontal axis illustrates the number of faces of different tooth models. The vertical axis illustrates DCD.

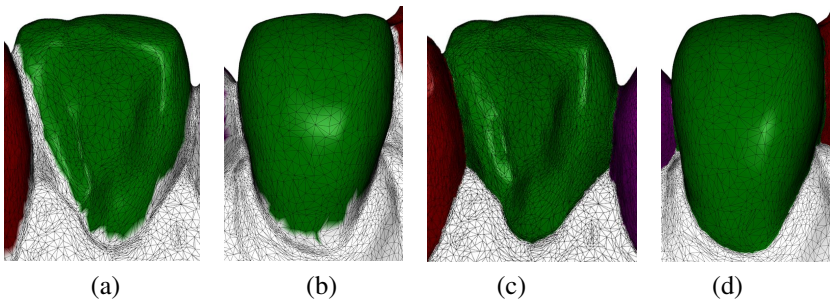


Figure 11: (a)&(b) are different views of the segmentation result of FCC method. (c)&(d) are different views of our improved segmentation results.

vertices and faces vary greatly. The results in Figure 9 and Figure 10 show that the segmentation accuracy and boundary errors are related to the face number of the model. The results in the red box show the results with lower accuracy and larger errors. It can be seen that the models with higher segmentation accuracy and smaller boundary errors tend to have more faces. Generally, the more vertices and faces the model has, the higher quality the model reconstruction is. At the boundary of two objects, the density of vertices and faces is higher than that of flat areas, and the negative curvature feature is more obvious. Due to the low accuracy of some 3D scanning devices, the negative curvature feature is not obvious enough, which leads to slightly worse segmentation results.

5. Conclusions

This paper presents an algorithm for tooth model segmentation in a harmonic parameter space. This method takes a 3D tooth model as input and outputs the label of each vertex of the mesh. Our method first maps the 3D tooth model isomorphically into the 2D harmonic parameter space, and then projects the 2D plane mesh to a 256×1024 image. Following the U-Net structure, we designed the convolutional neural network to train a highly robust tooth image segmentation model, which can take the tooth image as input and obtain the corresponding segmentation mask. Finally, we map the image segmentation mask back to the 3D tooth model, and improve the FCC algorithm to refine the segmentation, so as to get an accurate and smooth segmentation boundary. Our average segmentation accuracy is 98.87%, achieving state-of-the-art, which can prove the effectiveness of our method. Our method has been applied to a commercial orthodontic CAD system, and achieves satisfactory performance in practice.

Our method still has some limitations. First of all, our method requires the tooth model to be a non-closed genus-zero 3D surface with only one boundary to satisfy the input conditions of mesh parameterization. Therefore, a tedious pre-processing operation is required before the tooth segmentation. Secondly, the error of the segmentation mask predicted by the neural network should not be too large, otherwise it is difficult to find an accurate boundary even after the segmentation refinement step. If the size of noise area (in the red box of Figure 7) in the predicted segmentation mask is too large, it will lead to the failure of denoising process. If the prediction masks of adjacent teeth are connected to each other, it will cause these teeth to be labeled as one tooth. So our future work is to design a refined subnet to deal with the prediction of the boundary of the mask and remove the noise area. Thirdly, The final segmentation accuracy still heavily depends on the max-flow algorithm. Due to the low quality of some models, the final segmentation boundaries for these models are still rough and the errors are larger, so we plan to add boundary smoothing conditions to the segmentation refinement step to find more accurate and smooth boundaries. Finally, we have a limited data set with only 120 tooth models. Although we designed cross-validation and comparison experiments to prove the reliability of our method, it is necessary to expand the data set to make our method more reliable.

Acknowledgements

This work was supported by Science and Technology Service Network Initiative (no. KFJ-STS-ZDTP-070 and KFJ-STS-QYZD-129), National Key R&D Program of China (no. 2018AAA0103002), Royal Society Newton Advanced Fellowship (no. NAF\R2\192151) and the Open Research Fund from Shenzhen Research Institute of Big Data (no. 2019ORF01013).

[1] K. Wu, L. Chen, J. Li, Y. Zhou, Tooth segmentation on dental meshes using morphologic skeleton, Computers & Graphics 38 (2014) 199–211.

[2] S. M. Yamany, A. M. El-Bialy, Efficient free-form surface representation with application in orthodontics, in: Three-Dimensional Image Capture and Applications II, Vol. 3640, International Society for Optics and Photonics, 1999, pp. 115–125.

[3] T. Yuan, W. Liao, N. Dai, X. Cheng, Q. Yu, Single-tooth modeling for 3D dental model, International Journal of Biomedical Imaging 2010 (1) (2010) 9.

[4] Y. Kumar, R. Janardan, B. Larson, J. Moon, Improved segmentation of teeth in dental models, Computer-Aided Design and Applications 8 (2) (2011) 211–224.

[5] T. Kronfeld, D. Brunner, G. Brunnett, Snake-based segmentation of teeth from virtual dental casts, Computer-Aided Design and Applications 7 (2) (2010) 221–233.

[6] Z. Li, X. Ning, Z. Wang, A fast segmentation method for STL teeth model, in: IEEE/ICME International Conference on Complex Medical Engineering, 2007.

[7] C. Sinthanayothin, W. Tharanont, Orthodontics treatment simulation by teeth segmentation and setup, in: International Conference on Electrical Engineering/electronics, 2008.

[8] Y. Ma, Z. Li, Computer aided orthodontics treatment by virtual segmentation and adjustment, in: International Conference on Image Analysis & Signal Processing, 2010.

[9] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical Image Computing & Computer-assisted Intervention, 2015.

[10] G. Kan, D. Zou, X. Chen, 3D mesh labeling via deep convolutional neural networks, ACM Transactions on Graphics 35 (1) (2015) 1–12.

[11] X. Xu, C. Liu, Y. Zheng, 3D tooth segmentation and labeling using deep convolutional neural networks, IEEE transactions on visualization and computer graphics 25 (7) (2018) 2336–2348.

[12] M. Eck, T. DeRose, T. Duchamp, H. Hoppe, M. Lounsbery, W. Stuetzle, Multiresolution analysis of arbitrary meshes, in: ACM SIGGRAPH, 1995, pp. 173–182.

[13] S. Katz, A. Tal, Hierarchical mesh decomposition using fuzzy clustering and cuts, ACM Transactions on Graphics 22 (3) (2003) 954–961.

[14] R. Ahlswede, N. Cai, S. Y. R. Li, R. W. Yeung, Network information flow, IEEE Transactions on Information Theory 46 (4) (2000) 1204–1216.

[15] A. Shamir, A survey on mesh segmentation techniques, Computer Graphics Forum 27 (6) (2010) 1539–1556.

[16] S. Shlafman, A. Tal, Metamorphosis of polyhedral surfaces using decomposition, Computer Graphics Forum 21 (3) (2010) 219–228.

[17] T. G. Debelee, F. Schwenker, S. Rahimeto, D. Yohannes, Evaluation of modified adaptive k-means segmentation algorithm, Computational Visual Media (2019) 1–15.

[18] G. Lavou, F. Dupont, A. Baskurt, A new CAD mesh segmentation method, based on curvature tensor analysis, Computer-Aided Design 37 (10) (2005) 975–987.

[19] M. Attene, B. Falcidieno, M. Spagnuolo, Hierarchical mesh segmentation based on fitting primitives, Visual Computer 22 (3) (2006) 181–193.

[20] A. P. Mangan, R. T. Whitaker, Partitioning 3D surface meshes using watershed segmentation, IEEE Transactions on Visualization and Computer Graphics 5 (4) (1999) 308–321.

[21] Y.-K. Lai, S.-M. Hu, R. R. Martin, P. L. Rosin, Fast mesh segmentation using random walks, ACM Symposium on Solid & Physical Modeling (2008) 183–191.

[22] A. Golovinskiy, T. A. Funkhouser, Randomized cuts for 3D mesh analysis, ACM Transactions on Graphics 27 (5) (2008) 1–12.

[23] S. Katz, G. Leifman, A. Tal, Mesh segmentation using feature point and core extraction, Visual Computer 21 (8-10) (2005) 649–658.

[24] L. Shapira, A. Shamir, D. Cohen-Or, Consistent mesh partitioning and skeletonisation using the shape diameter function, Visual Computer 24 (4) (2008) 249.

[25] Y. Lee, S. Lee, A. Shamir, D. Cohen-Or, H. P. Seidel, Intelligent mesh scissoring using 3D snakes, in: Conference on Computer Graphics & Applications, 2004.

[26] Y. Lee, S. Lee, A. Shamir, D. Cohen-Or, H. P. Seidel, Mesh scissoring with minima rule and part salience, Computer Aided Geometric Design 22 (5) (2005) 444–465.

[27] L. Yin, K. Guo, B. Zhou, Q. Zhao, 3D shape co-segmentation via sparse and low rank representations, Science China Information Sciences 61 (5) (2018) 054101.

[28] Z. Ji, L. Liu, Z. Chen, G. Wang, Easy mesh cutting, in: Computer Graphics Forum, 2006, pp. 283–291.

[29] L. Fan, L. Liu, K. Liu, Paint mesh cutting, in: Computer Graphics Forum, 2011.

[30] Z. Youyi, T. Chiew-Lan, A. Oscar Kin-Chung, Dot scissor: a single-click interface for mesh segmentation, IEEE Transactions on Visualization & Computer Graphics 18 (8) (2012) 1304.

[31] M. Min, L. Fan, L. Liu, ICutter: A direct cut-out tool for 3D shapes, Computer Animation & Virtual Worlds 22 (4) (2011) 335–342.

[32] Y. Zheng, C. L. Tai, Mesh decomposition with cross-boundary brushes, Computer Graphics Forum 29 (2) (2010) 527–535.

[33] O. K. Au, Y. Zheng, M. Chen, P. Xu, C. L. Tai, Mesh segmentation with concavity-aware fields, IEEE Transactions on Visualization & Computer Graphics 18 (7) (2012) 1125.

[34] D. Khan, D.-M. Yan, F. Ding, Y. Zhuang, X. Zhang, Surface remeshing with robust user-guided segmentation, Computational Visual Media 4 (2) (2018) 113–122.

[35] Y. Zhuang, M. Zou, N. Carr, T. Ju, Anisotropic geodesics for live-wire mesh segmentation, Computer Graphics Forum 33 (7) (2014) 111–120.

[36] X. Chen, A. Golovinskiy, T. A. Funkhouser, A benchmark for 3D mesh segmentation, ACM Transactions on Graphics 28 (3) (2009) 1–12.

[37] E. Kalogerakis, A. Hertzmann, K. Singh, Learning 3D mesh segmentation and labeling, in: ACM SIGGRAPH, 2010.

[38] Y. Wang, S. Asafi, O. V. Kaick, Z. Hao, B. Chen, Active co-analysis of a set of shapes, ACM Transactions on Graphics 31 (6) (2012) 157:1–157:10.

[39] H. Benhabiles, G. Lavou, J. P. Vandeborre, M. Daoudi, Learning boundary edges for 3D-mesh segmentation, Computer Graphics Forum 30 (8) (2011) 2170–2182.

[40] Y. Lu, M. Zhen, T. Fang, Multi-view based neural network for semantic segmentation on 3D scenes, Science China Information Sciences 62 (12) (2019) 229101.

[41] M. Zhao, L. Ma, W. Tan, D. Nie, Interactive tooth segmentation of dental models, in: Conference: International Conference of the IEEE Engineering in Medicine & Biology Society IEEE Engineering in Medicine & Biology Society Conference, 2005.

[42] B. J. Zou, S. J. Liu, S. H. Liao, X. Ding, Y. Liang, Interactive tooth partition of dental mesh base on tooth-target harmonic field, Computers in Biology & Medicine 56 (C) (2015) 132–144.

[43] S. H. Liao, S. J. Liu, B. J. Zou, X. Ding, Y. Liang, J. H. Huang, Automatic tooth segmentation of dental mesh based on harmonic fields, Biomed Research International 2015 (3) (2015) 187173.

[44] Z. Li, H. Wang, Interactive tooth separation from dental model using segmentation field, PloS one 11 (8) (2016) e0161159.

[45] K. Toshiaki, S. H. Ong, K. W. C. Foong, Tooth segmentation of dental study models using range images, IEEE Trans. Med. Imaging 23 (3) (2004) 350–362.

[46] M. Grzegorzek, M. Trierscheid, D. Papoutsis, D. Paulus, A multi-stage approach for 3D teeth segmentation from dentition surfaces, in: International Conference on Image & Signal Processing, 2010.

[47] N. Wongwaen, C. Sinthanayothin, Computerized algorithm for 3D teeth segmentation, in: International Conference on Electronics & Information Engineering, 2010.

[48] W. T. Tutte, Convex representations of graphs, Proceedings of the London Mathematical Society 3 (1) (1960) 304–320.

[49] M. Foracchia, E. Grisan, A. Ruggeri, Luminosity and contrast normalization in retinal images, Medical Image Analysis 9 (3) (2005) 179–190.

[50] A. V. Goldberg, R. E. Tarjan, A new approach to the maximum-flow problem, Journal of the ACM (JACM) 35 (4) (1988) 921–940.