3D Tooth Segmentation and Labeling using Deep Convolutional Neural Networks

Xiaojie Xu[1], Chang Liu[1], and Youyi Zheng[†]

Abstract—In this paper, we present a novel approach for 3D dental model segmentation via deep Convolutional Neural Networks (CNNs). Traditional geometry-based methods tend to receive undesirable results due to the complex appearance of human teeth (e.g., missing/rotten teeth, feature-less regions, crowding teeth, extra medical attachments, etc.). Furthermore, labeling of individual tooth is hardly enabled in traditional tooth segmentation methods. To address these issues, we propose to learn a generic and robust segmentation model by exploiting deep Neural Networks, namely NNs. The segmentation task is achieved by labeling each mesh face. We extract a set of geometry features as face feature representations. In the training step, the network is fed with those features, and produces a probability vector, of which each element indicates the probability a face belonging to the corresponding model part. To this end, we extensively experiment with various network structures, and eventually arrive at a 2-level hierarchical CNNs structure for tooth segmentation: one for teeth-gingiva labeling and the other for inter-teeth labeling. Further, we propose a novel boundary-aware tooth simplification method to significantly improve efficiency in the stage of feature extraction. After CNNs prediction, we do graph-based label optimization and further refine the boundary with an improved version of fuzzy clustering. The accuracy of our mesh labeling method exceeds that of the state-of-art geometry-based methods, reaching 99.06% measured by area which is directly applicable in orthodontic CAD systems. It is also robust to any possible foreign matters on model surface, e.g., air bubbles, dental accessories, and many more.

Index Terms—Boundary-aware simplification, 3D mesh segmentation, deep convolutional neural networks, fuzzy clustering.

1 Introduction

As computers developed, computer-aided-design(CAD) systems appear in more and more fields. They take advantage of hardware-supported computer graphics technology to effectively and efficiently do the tasks which are traditionally of high labour intensity. Most of the dental clinics around the world use CAD systems to develop treatment plans, e.g., orthodontics. Orthodontic CAD systems play an important role in modern dentistry. They accept a three dimension (3D) dental model, specified by patients' own impression, as input and assist dentists to extract, move, delete, and rearrange teeth for simulating the treatment's outcome. With an automatic processing system, dentists will be set free from the time-consuming and boring task.

Tooth segmentation and labeling is the most fundamental and critical component of these CAD systems, which remains unsolved. The major challenges are as follows. As a part of human body, teeth, similar to fingerprints, vary from one person to another. There is no deterministic parametric description to cover any individual tooth of all people. Besides, dental models from patients always suffer from strange abnormalities, for example, the crowding problem, which is described as the two neighbouring teeth are misaligned, hence the boundary between them is implicit, which results in the disappearance of normal interstices. Furthermore, missing/rotten teeth and holes are commonly seen among people, which bring in additional challenges. These properties play the devil with traditional geometry-based methods. Curvature-based methods tend to divide a surface into several parts along the concave discontinuity of the tangent plane, and thus, are not reliable towards feature-less regions with smooth varying curvatures, e.g., in the lingual portion of the tooth [1]. Another challenge is the noise generated during the scanning of plaster models, such as air bubbles and the inaccuracy of plaster models, which usually happen deep in the mouth, around the wisdom teeth area. Moreover, some patients may wear dental accessories when making the radiology. Such foreign matters disturb the feature distribution of each individual tooth, thus have an adverse impact on the segmentation task.

Due to these challenges, traditional geometry-based methods are less suitable for tooth segmentation task in practice, as they are lack of robustness to complex tooth shapes and tooth arrangements. Besides, other existing image-based or interactive methods [2], [3], [4] are either labour-intensive or not accurate enough, which makes the development of an automatic, generic, and accurate tooth segmentation framework demanding.

In this paper, we propose a data-driven method for 3D dental mesh segmentation. In particular, we exploit deep Convolutional Neural Networks(CNNs) model for the task of tooth segmentation. The network is designed for labeling each tooth triangle. We extract 600-dimension geometry features for each mesh face and pack them into a 20×30

[1] *co-first author*
[†] *corresponding author.*

- *X. Xu and C. Liu are with Chinese Academy of Sciences, Shanghai Institute of Microsyst & Information Technology, Shanghai 200050, Peoples R China. And they are also with ShanghaiTech University, School of Information Science & Technology, Shanghai 201210, and University of Chinese Academy of Sciences, Peoples R China.*
  E-mail: xuxj,liuchang1@shanghaitech.edu.cn
- *Y. Zheng is with the State Key Lab of CAD&CG, Zhejiang University, Hangzhou, China, 310058.*
  E-mail: zyy@cad.zju.edu.cn

image as in [5]. By doing so, we get a sequence of image-label pairs as CNNs input. Usually, 50% of triangle faces on a dental mesh belong to gingiva, others belong to 14˜16 teeth. Besides, the quality of the segmentation boundary is vital in the subsequent orthodontic treatments (e.g., root planning). To deal with data imbalance and improve boundary accuracy, we extensively experiment with various network structures and finally arrive at a hierarchical labeling architecture, which consists of two CNNs, one for teeth-gingiva labeling (we call it TGCNNs), the other for inter-teeth labeling (we call it TTCNNs). As raw dental models used in orthodontic treatments are usually very large (200,000˜400,000 triangles) which poses a large overhead on the feature extraction step (up to dozens of hours). To improve the efficiency of the feature extraction while leaving the quality of CNNs unaffected, we design a boundary-aware mesh simplification algorithm and a correspondence-free mapping algorithm to pre-process and post-process the dental meshes.

For a newly come dental mesh S, which is to be labeled, we first simplify it to a mesh S' using our boundary-aware simplification algorithm. Then we extract the feature images per face on S' and feed them into the TGCNNs net for teeth-gingiva separation. In an intermediate step, we do label optimization to smooth the labeling boundary. We then apply the TTCNNs to label each individual tooth face. After that, we do label optimization to smooth the labeling results again and back-project the labeling result to the original model S. Finally, we optimize the segmentation boundary via an improved fuzzy clustering to achieve the final result.

Our data-driven tooth labeling method is capable of segmenting various dental models regardless of their geometric variations. It is not only effective and efficient, but also, to the best of our knowledge, the most accurate tooth segmentation and labeling model in the literature so far which achieves a practical precision of 99.06%. The main contributions of our method are:

- A simultaneous and robust teeth segmentation and labeling framework which achieves 99.06% accuracy and can be directly applied in industrial orthodontic CAD systems;
- A carefully designed 2-level hierarchical CNNs model trained on $1,000$ dental meshes, which is robust and generalize well on new data;
- A boundary-aware mesh simplification method to enable efficient feature extraction;
- An improved fuzzy clustering boundary optimization algorithm coupling network prediction with geometry optimization.

2 Related Work

This paper proposes a data-driven method for dental mesh segmentation. We first review the literature on general mesh segmentation. Then, we discuss approaches for dental mesh segmentation methods. Finally, we briefly introduce recent data-driven shape analysis methods in geometry processing.

General Mesh Segmentation 3D mesh segmentation is a fundamental task for mesh understanding and processing. It divides 3D shapes into several parts under reasonable criteria. Common approaches are mentioned in surveys [6], [7]. These approaches rely on geometry information more or less. They can be grouped into two categories: region-based and boundary-based segmentation approaches. Region-based approaches attempt to partition meshes into several regions, where mesh faces share similar characteristics, while faces in different regions differ greatly. Well-known region-based works include K-means [8], clustering [9], decomposition [10], fitting primitives [11], watersheds [12], random walks [13], fast marching [14]. While boundary-based approaches concentrate on finding the optimal curves to separate two neighbouring parts. They determine final curves by maximizing the difference between parts. Main methods include normalized and randomized cut [15], core extraction [16], shape diameter function [17], and active contours or scissoring [18], [19]. However, geometry-based methods tend to fail when meshes become special and complicated.

As meshes vary from each other in terms of appearance, it is impossible to separate a mesh into desired parts with fully automatic approach. While manual segmentation is tiring as well as time-wasting, sketch-based semi-automatic methods become popular. They provide simple and user-friendly interfaces for users to add their suggestions as start points or optimization constraints. Literature [20], [21] briefly described numbers of sketch-based segmentation methods. For example, Ji et al. [22] introduced an improved region-growing algorithm for segmentation. Fan et al. [23] adopted an efficient local graph-cut-based optimization algorithm and received satisfying results. Studies [24], [25], [26], [27] integrated harmonic field theory with sketch-based segmentation, which possess solid theoretical basis and work well. However, sketch-based methods require a balance between user input and automatic computation.

Since 3D mesh databases, e.g., the Princeton Segmentation Benchmark [28], were released, data-driven methods have been proposed for mesh segmentation. Both non-supervised and semi-supervised learning methods try to learn a model for separating a mesh meaningfully from the database and verify them on new meshes. Some recent works include [29], [30], [31].

Dental Mesh Segmentation Numerous segmentation approaches have been proposed to separate dental models. According to the input format, we divide the existed approaches into two categories, 2D image and 3D mesh.

Researchers have proposed effective segmentation algorithms based on the 2D projection images. Yamany et al. [2] encoded the curvature and surface normal information into a 2D image, and designed an image segmentation tool to extract structures of high/low curvature. By iteratively removing these structures, individual teeth surfaces are obtained. Similarly, Kondo et al. [32] presented an automated method for tooth segmentation from 3D digitized image captured by a laser scanner. Grzegorzek et al. [33] presented a multi-stage approach for tooth segmentation from 3D dentition surfaces based on a 2D model-based contour retrieval algorithm. Wongwaen et al. [34] converted the 3D-panoramic to 2D space to find the cutting points for segmentation of individual tooth, followed by converting the 2D image back to 3D space for remaining operations.

Literature [3] subdivided those methods, which take 3D mesh as input, into 3 categories. The first is curvature-

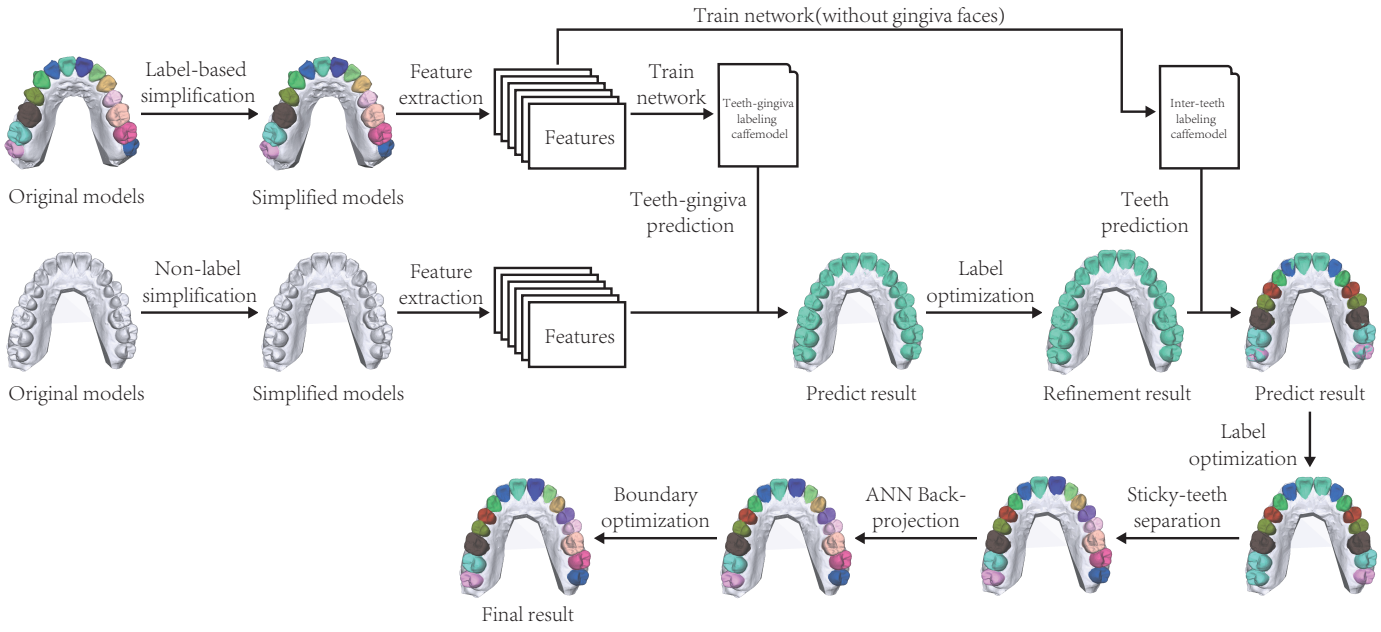


Fig. 1. The pipeline of our method. Our method takes a raw teeth model as input, simplifies the model, extracts its features and feeds into a 2-level hierarchical network to generate a label prediction, followed by label optimization and back-projection to get the final segmentation result.

based method, which separates the dental models relying on surface curvature. Yuan et al. [35] analysed the regions of the 3D dental model and classify them based on the minimum curvatures of the surface. Zhao et al. [4] proposed an interactive segmentation method based on curvature values of the triangle mesh. System designed by [36] requires users to provide a one-time setting of a certain curvature threshold via an intuitive slider. Others, including snake-based active contour method [37], "fast marching watershed" method [38] and morphologic skeleton extraction method [1] are all related to curvature information to some extent.

The second is contour-line-based method, which is a relative accurate segmentation method as it allows human interaction. In studies [39] and [40], users assign the boundary between each tooth and gum in the form of surface points by mouse click. Then the algorithm connects each pair of the neighbouring points depending on the geodesic information. The generated segments are desired boundary. Although the boundary is accurate, this method relies too much on user interaction. Users have to rotate/translate the mesh and zoom in/out repetitively to make their suggestions, which is tiring and time-consuming.

The third is harmonic-field-based method. Zou et al. [3] proposed a harmonic-field-based segmentation method which requires only a limited number of surface points as prior. It saves users' time and achieves reasonable results.

Data-driven Shape Analysis Recently, data-driven geometry processing algorithm has been developed both in computer graphics and computer vision communities [41]. The commonly known shape analysis techniques can be grouped into several topics, such as classification [42], [43], [44], matching [45], [46], [47], reconstruction [48], [49], [50], modeling and synthesis [51], [52], [53]. Data-driven segmentation methods are classified into supervised [54], [55], [56], semi-supervised [30], [57], [58] and unsupervised [58], [59], [60]. In order to design a data-driven algorithm, sufficient shape databases are necessary. Famous databases include [28], [61] which are maintained by universities and [58] which is collected from the web. Another way to gather adequate shape models is to create synthesis dataset, e.g., [62]. Xu et al. [63] made a comprehensive survey on the existing online data collections.

Traditional learning methods are mainly designed for finding a different representation of 3D mesh. In recent years, deep neural networks show their excellent performance in extracting latent features, as well as automatically building mappings between input and output [64], [65]. Especially, deep CNNs do well in image-format-input tasks [66], [67]. Researchers in computer graphics community are making efforts to feed 3D mesh data into CNNs. Guo et al. [5] extracted a 20×30-dimension geometric feature image for each triangle face and feed it into a typical classification network together with the ground-truth face label. Maron et al. [68] parameterized the sphere-like mesh to get a 2D image and use it to train a modified FCN-like [69] network. [5], [68], [70] show that if well designed, CNNs are also capable of 3D mesh segmentation.

3 Overview

Fig.1 illustrates our pipeline. Due to the computational burden of features for large meshes [29], we do mesh simplification for each dental model to reduce the face numbers. To account for the preservation of informative geometric features for segmentation, we design a boundary-aware mesh simplification algorithm to maintain the features along the teeth-gingiva and tooth-tooth regions.

We extract global and local features of each face on the simplified model. We use the similar set of features as in [29], and add positional features to boost the network performance. These features are reorganized into a 20×30

image to feed into the network. We design a 2-level hierarchical network for face labeling. We train two CNNs with similar layers for teeth-gingiva and inter-teeth labeling respectively. The CNN architecture consists of convolution, pooling and fully-connected layers, with carefully tuned parameters, e.g., the number of layers and the activation functions.

Right after each network prediction, we employ label optimization to correct the wrongly predicted labels which usually appear on the boundary. We further improve the boundary between teeth and gingiva as well as between individual teeth by graph optimization. We also employ PCA analysis to split sticky-teeth (i.e., pairs of teeth which are adjacent and get the same label after optimization) which occasionally appear in regions with missing/rotten teeth or the front teeth. Finally, we back-project the labels of simplified model onto the original model and further refine the boundary. Below we discuss various algorithmic design choices in detail.

4 ALGORITHM

4.1 Boundary-aware Tooth Simplification

A dental model, acquired by CT scanning, is very precise, containing more than $200,000$ faces. A direct computation of geometric features [29] on such fine model is extremely time-consuming in either training or testing stage (Section 4.2). Mesh simplification, therefore, is necessary for pre-processing dental meshes. Traditional feature-preserving mesh simplification methods tend to lose semantic information, e.g., details on the teeth-gingiva boundary. Clear and accurate tooth-tooth, together with teeth-gingiva boundary plays an important role in the learning procedure (Section 5). Thus we design a boundary-aware tooth simplification algorithm to preserve such semantic information as much as possible.

To preserve the boundary information, first we need to identify them. Our aim is to divide a dental model into three regions: gingiva, teeth, and teeth-gingiva boundary, shown in Fig.2d. The gingiva region occupies a large part of triangle faces but provides little discriminative information for classification as it is feature-less. The teeth region possesses more important geometric details than gingiva, and should not be simplified too much. The boundary region is the most important for segmentation, whose details should be retained as much as possible. To this end, we modify the traditional mesh simplification method [71]. We multiply the edge-collapse-cost with different weights in different regions.

Our task now is to identify those regions. We observe that the dental meshes are usually scanned on the same CT platform, whose bottom parts are planar, as shown in Fig.2a. This makes the classification task easier. We first identify the largest plane using a greedy floodfill algorithm and align the normal of the largest plane with the z-axis. It is easy to align x-axis and y-axis by PCA analysis. Then for an upper-part dental mesh, shown in Fig.2, the majority of teeth faces appear in regions which are of larger z-axis coordinate values, far from the mesh center, and close to teeth sharp points [3], i.e., the red points in Fig.2a. For teeth-gingiva boundary, it is usually continuous, i.e., the green points in Fig.2a.

We denote a dental mesh as $\mathcal{G} = \langle \mathcal{V}, \mathcal{E}, \mathcal{F} \rangle$, where $\mathcal{V}$ is the vertex set, $\mathcal{E}$ edge set, and $\mathcal{F}$ face set. Label set $\mathcal{L}$ consists of the label $l_i \in [0, 1]$ of each triangle face, where 0 represents definite gingiva and 1 represents definite teeth. Labels $\mathcal{L}$ are determined by optimizing

$$\arg \min_{\{l_i, i \in \mathcal{F}\}} \sum_{i \in \mathcal{F}} E_1(l_i) + \lambda \sum_{i,j \in \mathcal{F}} E_2(l_i, l_j) \tag{1}$$

where λ is a non-negative constant to balance the two terms (empirically set $\lambda = 100$).

The unary term is defined as

$$E_1(l_i) = \alpha_1 E_{u1}(l_i) + \alpha_2 E_{u2}(l_i) + \alpha_3 E_{u3}(l_i) \quad s.t. \quad \alpha_1 + \alpha_2 + \alpha_3 = 1 \tag{2}$$

where E_{u1}, E_{u2}, E_{u3} are the probability energy given by the three characteristics mentioned before, which are z-axis coordinate, geodesic distance to the nearest sharp points and Euclidean distance to mesh center in XY plane. Specifically, each item is defined as

$$\begin{cases} E_{u1}(l_i = 0) = (z_i - z_{\min})/H \\ E_{u2}(l_i = 0) = 1 - gd_i / gd_{\max} \\ E_{u3}(l_i = 0) = \sqrt{(\frac{x_i - x_{mesh}}{0.5L})^2 + (\frac{y_i - y_{mesh}}{0.5W})^2} \\ E_1(l_i = 1) = 1 - E_1(l_i = 0) \end{cases} \tag{3}$$

where x_i, y_i, z_i are the x, y, z-axis coordinate of triangle face i. $[L, W, H]$ are the length (in the x-axis direction), width (in the y-axis direction) and height (in the z-axis direction) of the axis-aligned bounding box of a dental model. gd_i is the geodesic distance from face i to the nearest sharp point and $gd_{\max}$ is the maximum value for all gd_i. x_{mesh} and y_{mesh} are the x, y-axis coordinate of mesh center (set $\alpha_1 = 0.4, \alpha_2 = 0.5, \alpha_3 = 0.1$). We detect the sharp feature points as local shape extremities [3]. The probability field is shown in Fig.2b.

Faces on teeth-gingiva boundary usually have negative curvature. We use pairwise term E_2 to measure this as

$$E_2(l_i, l_j) = \begin{cases} \dfrac{1}{1 + \frac{AD(\alpha_{ij})}{avg(AD)}}, & l_i \neq l_j \\ 0, & l_i = l_j \end{cases} \tag{4}$$

The angular distance is $AD(\alpha_{ij}) = \eta(1 - \cos \alpha_{ij})$, where α_{ij} is the angle between the normal of face i and j. The definition here is the same as [10]. For convex angles, $\eta = 0.05$, and for concave angles $\eta = 1$.

We use graph cuts algorithm to solve the optimization problem in Eqn. 1. As a result, the mesh is divided into two regions: teeth and gingiva. We extend the teeth-gingiva boundary using Breadth First Search(BFS)(5 iterations in our experiments) to get three regions (Fig.2d). Then we shall conduct a detail-preserved mesh simplification (Fig.2c). Empirically, we set the collapsing weights for edges in gingiva, teeth, and gingiva-teeth boundary regions as 1, 20, and 500 respectively. The simplification ratio is 0.2. The simplified model has around $40,000$ triangle faces.

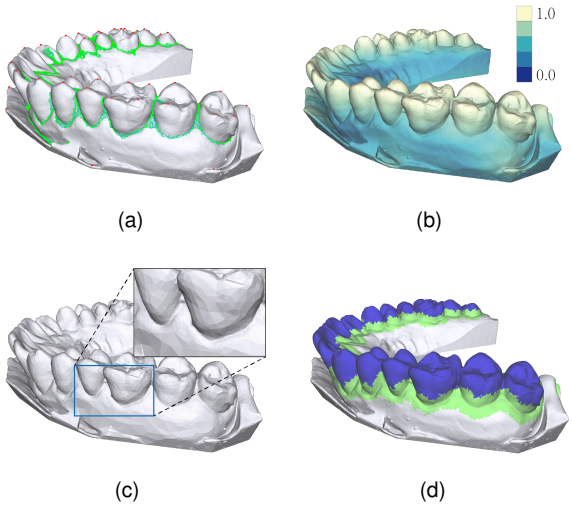


Fig. 2. (a) Sharp points are colored in red and negative curvature points are colored in green. (b) The probability field of belonging to the teeth. Light color indicates large probability. (c) Our simplification result. (d) Three regions. After optimizing Eqn. 1 by graph cuts, we roughly divide the model into teeth and gingiva regions. Then we extend the teeth-gingiva boundary using BFS and get the three regions, i.e., gingiva(gray), tooth(blue), the area near teeth-gingiva boundary(green).

4.2 Feature Extraction

After mesh simplification, we extract 8 different types of geometric features, which form a 600-dimension vector for each triangle face. The feature vector is then reorganized into a 20×30 feature image, so that it can be fed into CNNs. These geometric features include curvature(CUR) [72], PCA feature(PCA) [29], shape context(SC) [73], shape diameter function(SDF) [74], spin image(SI) [75] and coordinates(COORD). Other than the coordinates, others are the same as those used in [5][1]. These features are computed under several scales to capture both local and global information. They make up the first 593 dimensions of the feature vector. As for the last 7 dimensions, we introduce coordinate features based on the natural distribution of dental models. For example, incisors are always found in a specified relative location of all dental models. The coordinate feature consists of 3D Cartesian coordinates (x, y, z) (after alignment as in Section 4.1), spherical coordinate (ρ, θ, ϕ) and the absolute value of ϕ.

4.3 Networks Architecture

In recent years, CNNs have gone through rapid development. Traditional CNNs have become deeper and wider to learn feature as much as possible from the large-scale training data. Besides, specially designed network structures are proposed for high level computer vision tasks. However, basic CNN structures are sufficient for studies like [5]. Such CNNs possess clear structures with fewer parameters to train. The training complexity decreases a lot. Therefore, we design a CNN architecture based on [5] for dental mesh segmentation. Notice that the CNNs proposed by [5] is a typical architecture for classification task, each face is set with the most probable class label. We exactly follow this key idea.

1. Source code can be found at http://people.cs.umass.edu/~kalo/papers/LabelMeshes/.

4.3.1 Hierarchical mesh labeling

A dental model (upper or lower part) consists of gingiva and sixteen teeth, however, nearly half of the triangle faces belong to gingiva, while others are subdivided into sixteen groups. This imbalanced label distribution possesses difficulties in the network training. We experimented with a few design variants (refer to Section 5), and found the following 2-level hierarchical network architecture achieves the best performance, as shown in Fig.1.

The first level of our hierarchy is to conduct teeth-gingiva separation. We train a 2-label classification network. The second level is to separate each tooth. Taking the symmetry of dental models into consideration, we train an 8-label classification network. We later use geometric information to separate the left and right part and finally get a dental model with seventeen labels. The structure of the two networks are identical, as shown in Table 1.

4.3.2 Networks structure

Our network takes the reorganized 20×30 feature image, denoted as $\mathbf{X}^0$, generated in Section 4.2, as input and outputs face label $\mathcal{L} = \{0, 1\}$ or $\mathcal{L} = \{0, \cdots, 7\}$. We divide the network layers into three blocks: two convolution blocks (CB) and one fully-connected block (FB), see Table 1. In *CB1*, the convolution layer *conv1* is applied to $\mathbf{X}^0$ as

$$\mathbf{Y}_i^l = \mathbf{W}_i^l \otimes \mathbf{X}^{l-1} + \mathbf{b}_i^l, \quad i = 1, \ldots, 16, \quad l = 1, \tag{5}$$

where $\otimes$ indicates the convolution operation. The bias $\mathbf{b}_i^l$ is uniformly added to each pixel of $\mathbf{W}_i^l \otimes \mathbf{X}^{l-1}$. The output $\mathbf{Y}_i^l$ then goes through an activation function, called parametric ReLU (pReLU)

$$f(x) = \begin{cases} y_i, & y_i > 0 \\ a_i y_i, & y_i \leq 0 \end{cases} \tag{6}$$

where a_i is a trainable parameter. Then we achieve sixteen feature maps $\{\mathbf{X}_i^1\}_{i=1}^{16}$, which is computed by

$$\mathbf{X}_i^l = f(\mathbf{Y}_i^l), \quad l = 1 \tag{7}$$

For *conv2*, we figure out $\{\mathbf{X}_i^2\}_{i=1}^{32}$ by setting $l = 2$ in Eqn. 5 and Eqn. 7. After that, we feed the thirty-two feature

TABLE 1
Network structure.

	layer	parameters
CB1	conv1	3×5, 16
	conv2	3×3, 32
	pool1	2×2, max
CB2	conv3	3×3, 64
	conv4	3×3, 128
	pool2	2×2, max
FB	fc1	100
	dropout	0.5
	fc2	2(or 8)
	softmax	

maps to pooling layer *pool1*, which picks up the maximum value of each 2×2 non-overlapping patch to represent it. The feature maps output by *pool1*, noted as $\{\widehat{\mathbf{X}}_i^2\}_{i=1}^{32}$ is $1/4$ of the input size.

The block *CB2* has similar composition with *CB1*. $\{\widehat{\mathbf{X}}_i^2\}_{i=1}^{32}$ acts as the input of *conv3*. We reorganize the output of *pool2*, denoted as $\{\widehat{\mathbf{X}}_i^4\}_{i=1}^{128}$, into a vector to feed into fully-connected layer *fc1*. In order to simplify the network's complexity and reduce over-fitting, we randomly drop out 50% of the layer nodes during each training iteration. Then, *fc2*, together with *softmax* layer generate the probability vector of length 2 (or 8).

4.4 Label Optimization

CNNs prediction generates a label and a probability vector for each face on the testing mesh. The prediction results are rough and inaccurate on the boundary. Small fragments appear where they are not supposed to be (Fig.3a, 3c). To fix this, we adopt the multi-label graph cuts method [76] to refine the prediction result after each network prediction.

Triangle face i is labeled by CNNs with l_i under the probability of p_i. The neighbouring faces of i are denoted as $\mathcal{N}_i$. Label optimization problem is solved by optimizing

$$\arg \min_{\{l_i, i \in \mathcal{F}\}} \sum_{i \in \mathcal{F}} \xi_U(p_i, l_i) + \lambda \sum_{i \in \mathcal{F}, j \in \mathcal{N}_i} \xi_S(p_i, p_j, l_i, l_j) \quad (8)$$

where λ is a non-negative constant ($\lambda = 20$ for teeth-gingiva classification, $\lambda = 100$ for inter-teeth classification). The first term is defined as $\xi_U(p_i, l_i) = -\log(p_i(l_i))$. The penalty ξ_U rises when probability $p_i(l_i)$ drops. For a dental mesh, the teeth-gingiva boundary tends to be concave, so we define the second term as

$$\xi_S(p_i, p_j, l_i, l_j) = \begin{cases} 0, & l_i = l_j \\ -\log(\frac{\theta_{ij}}{\pi})\phi_{ij}, & l_i \neq l_j, \theta_{ij} \text{ is concave} \\ -\beta_{ij}\log(\frac{\theta_{ij}}{\pi})\phi_{ij}, & l_i \neq l_j, \theta_{ij} \text{ is convex} \end{cases} \quad (9)$$

with $\beta_{ij} = 1 + |\hat{n}_i \cdot \hat{n}_j|$ and $\phi_{ij} = \|c_i - c_j\|_2$, where $\hat{n}_i$ is the face normal of triangle i, c_i is the barycenter of i, and θ_{ij} is the dihedral angle between i and j. We add the term β_{ij} to enforce the optimization to favor concave regions as the teeth-gingiva and tooth-tooth boundaries are usually concave. After label optimization, these small fragments will be removed, shown in Fig.3b,3d.

Handling sticky cases. Due to the fact that teeth vary among peoples in shape, size, number of missing parts and holes, the hierarchical structure occasionally leads to incorrect predictions around missing/rotten teeth. Besides, the central incisors are of the same label inherently (Fig.3d). In these cases (it happens at the rate of 7.72%, statistics collected from 150 test cases), the multi-label graph cuts algorithm labels two or more nearby teeth as the same. We take special treatments to these problems.

After inter-teeth segmentation, to distinguish one tooth from the other, we do PCA analysis for the classified teeth. Because teeth in our dataset have no root, the height might vary across models. We leave out the height axis which is normally aligned with z-axis and only consider the breadth axis of the teeth which is more reliable in measuring the change.

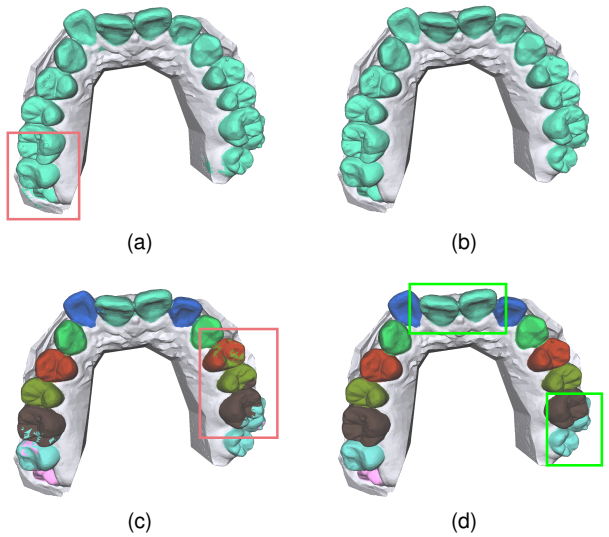


Fig. 3. (a) Teeth-gingiva CNNs prediction. (b) Teeth-gingiva refinement after label optimization. (c) Inter-teeth classification with eight labels after CNNs prediction. (d) Inter-teeth refinement after label optimization.

For each tooth, if its longest breadth axis is longer than a constant τ_1 (which is set as 1.4 times the mean value per tooth, calculated by doing PCA analysis on 1000 training data), we should break it into two teeth and repeat the processes if needed. This is regarded as another graph cuts problem, and can be solved by the similar labeling optimizing procedure as above. We refer to the appendix for details.

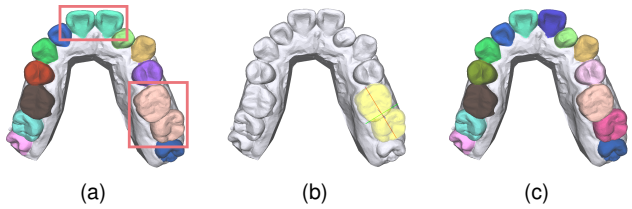


Fig. 4. (a) Sticky-teeth. (b) PCA analysis. (c) Separated teeth.

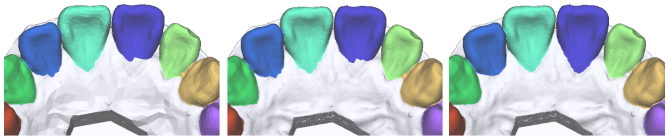
4.5 ANN Mapping

After labeling the simplified mesh properly, we need to project the results back to the original detailed model. As the two meshes are coordinate-aligned, we employ ANN[2] mapping, followed by additional label optimization (Eqn. 8) on the boundary, shown in Fig.5. It should be noticed that the probabilities from CNNs prediction are projected at the same time.

4.6 Boundary Smoothing

Accurate and smooth boundary is very important in dental treatment, as it will affect further processing, such as virtual gingiva generation, tooth rearrangement and dental appliance production. However, neither geometric information

2. ANN library can be found at https://www.cs.umd.edu/~mount/ANN/.



(a) (b) (c)

Fig. 5. (a) Segmentation results on simplified mesh. (b) The results on original mesh after ANN mapping(The boundary is not smooth). (c) The refinement results on original mesh after label optimization and boundary smoothing.

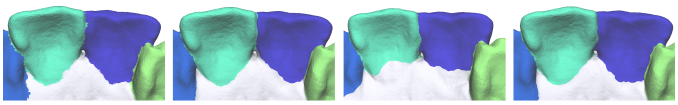
nor CNNs predictions are reliable to determine the optimal boundary by itself. We propose to combine them and use an improved fuzzy clustering algorithm [10] to refine the boundary. The improved fuzzy clustering method takes both geometry and CNNs predictions into consideration and works well on dental models.

For each tooth labeled by l, we first do BFS from the current boundary to visit a group of nearby faces, which make up the fuzzy region. For faces on the border of the region, we collect those faces adjacent to tooth l as set $\mathcal{S}$, and others as set $\mathcal{T}$. The modified capacity is

$$Cap_{new}(i,j) = \frac{1}{1+\exp(-\frac{x^2}{\sigma})} Cap(i,j) \tag{10}$$

$$Cap(i,j) = \begin{cases} \dfrac{1}{1+\frac{AD(\alpha_{ij})}{avg(AD)}}, & i,j \notin \mathcal{S},\mathcal{T} \\ \infty, & \text{else} \end{cases} \tag{11}$$

where x is the geodesic distance from the face center to the nearest current boundary($\sigma = 0.05$). $Cap(i,j)$ is identical to [10], and $AD(\alpha_{ij})$ is defined as before. The equation shows that faces close to the current boundary are of high probability to be the final boundary. In other words, we consider CNNs predictions as an important factor in determining the final location of boundaries. CNNs prediction is usually better than traditional geometry refinement towards feature-less (e.g., flat) regions (Fig.6). We further refine the boundary using a simple shortest-path dynamic programming, see appendix for details.



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

Fig. 6. (a) The results after ANN mapping. (b) Ground truth. (c) Boundary smoothing using fuzzy clustering and cuts [10]. As some boundary areas are very flat, it is hard to maintain CNNs prediction by directly applying [10]. (d) Our method shows respect to the CNNs prediction results.

5 Evaluation and Results

We conduct a number of experiments to show the effectiveness of our approach and validate various algorithmic components of our method, including mesh simplification, choice of features, network design, label optimization, and the boundary refinement. Our CNN models are trained with Caffe [77] on an i7-6700K CPU together with a GTX 1080 GPU.

Dataset. It is hardly possible to build up a dental model database with adequate amount for network training by ourselves. Luckily, a professional orthodontic company is willing to cooperate with us on this project. They provide us with manually labeled dental meshes, which can be regarded as the ground truth. We divide the dataset into 3 groups, shown in Table 2. All the results in this chapter are calculated on Test set.

TABLE 2
Dental mesh dataset.

Group	Training	Validation	Test
Number of Cases	1000	50	150

Metrics. We evaluate the performance from two different aspects. One is from a global perspective, i.e., to calculate the percentage of the area of correctly labeled faces [5], which is expressed as

$$Accuracy = \sum_{t\in\mathcal{T}} a_t \mathbf{g}_t(l_t) / \sum_{t\in\mathcal{T}} a_t \tag{12}$$

where a_t and l_t are the area and predicted label of triangle face t. $\mathbf{g_t}(l_t)$ is 1 if the prediction is correct, otherwise 0. As we have emphasized before, boundary accuracy is very important in tooth segmentation task. The other evaluation concentrates on the boundary among gingiva and teeth. We adopt the Directional Cut Discrepancy(DCD) [28] to evaluate the boundary mean errors. For simplicity, we denote upper tooth model as U, and lower tooth model as L in the following paragraphs.

Design choice of CNNs. As noted earlier, neural networks have been proved to be very effective for common classification tasks. There are many variants that could achieve similar results. Thus we explicitly explore different variants of network architectures to find the best match to our problem. The alternatives we have tried include LeNet [78], traditional Neural Networks (NNs) with modified deeper structure, 1-level CNNs with weighted loss function (WLF), which is commonly used on imbalanced datasets, and the locally connected graph autoencoder [79] (GR-DNN) which has been shown to be very effective in representation learning. For LeNet, we directly employ it on the 9-class teeth classification using our 20×30 feature as input. The structure of the second network, NNs, is shown in appendix, Fig.12a. The input is our 600-dimensional feature vector. In WLF, we set class weights for imbalanced data so that the 2-level structure is reduced to 1. Lastly, we implement GR-DNN and modify an intermediate layer of the original structure (from size 2 to size 100) to account for a proper embedding learning as our feature size is much larger. We also modify the softmax layer for the final 9-class classification (see the structure in Fig.12b).

The performance of these alternatives are shown in Table 3. Interestingly, all these alternatives have similar performance and all are able to get good results for tooth segmentation. However, our method achieves the best performance. We suspect that a wrap into images allows the network to seek more potential unknown relations among the extracted features than a single vector could do and a device of two

stages allows for more comprehensive fine tuning, for example, the label optimization parameter λ. As for GR-DNN, we tried two versions of the anchor graph [79], one is to create the anchor graph directly using the 600-dimensional feature space and the other is to use the Euclidean distance among the faces to account for local connectivity. However, in both cases we do not observe much difference, which indicates learning a better embedding does not necessarily guarantee a better classification result.

To evaluate the effectiveness of our pipeline, we make a comparison with method [5], which performs very well in general mesh segmentation. We prepare training data in the same way for the two methods, using mesh simplification followed with feature extraction. In the first comparison, we compare the two methods as themselves. The differences lie in the network structure, simplification and optimization methods. We use their own features, network and post-optimization. The output label numbers of the network proposed by [5] is set to seventeen, with the purpose of segmenting all teeth and gingiva at once. After network prediction, [5] only uses $\alpha-\beta$ swap to do label optimization. Table 3 shows that on large-scale training set, our method outperforms [5] significantly. In the second comparison, we verify the usefulness of the COORD feature, that is, we add a 7-dimension coordinate feature mentioned in 4.2 to [5] and conduct 17-label classification. The performance rises significantly (Table 3), which reveals the effectiveness of our new feature. This is because teeth are usually aligned and symmetry thus nearby teeth could confuse the network in the original feature representation. With the coordinate features, it could largely help distinguish for example, left and right.

Fig.11 shows some of our representative results. Note that our method is robust to various complex circumstances in human teeth such as missing/rotten teeth, irregular teeth arrangements, noise, bubbles, foreign attachments, as well as feature-less regions, thanks to our well designed network and the improved boundary refinement algorithm. Our networks are also very efficient, for a tooth model with $40,000$ triangles, it takes less than 1s for prediction. The simplification, ANN mapping, and fuzzy refinement take around 5s. The most time consuming step is the feature extraction, which takes 5 minutes per model. However, it is significantly faster than that on a raw model, i.e., if we do not use mesh simplification (which is 12 hours per model).

Verification of the effectiveness of boundary-aware simplification. To prove that the proposed boundary-aware simplification is effective, we make two sets of simplified models(on simplification ratio 0.2). One is simplified using boundary-aware algorithm, the other is simplified uniformly. Then we extract features to build up two sets of training data, 200 models and 1000 models. The networks are trained on two different amount of training data, in order to get rid of the influence of data scale. Results are shown in Table 4.

To keep consistent with training data, all test data are simplified under the same rule. Rows named TGCNNs and TTCNNs are the result of CNNs prediction. *Final* row refers to the final results of the whole pipeline. Judging by the numerical value, boundary-aware simplification is slightly higher than uniform one. To be honest, the difference between each (boundary-aware, uniform) pair is a bit trivial to evidence the effectiveness of boundary-aware algorithm. This is true as the boundary regions are indeed a small portion of the whole model. However, the visualization of these results supports that boundary-aware algorithm indeed helps networks predict much better towards boundary region. Fig.7 indicates that our boundary-aware simplification algorithm effectively prevents over-prediction and under-prediction in boundary regions. Such clean and accurate boundaries could largely benefit the subsequent processes in digital tooth treatments, for example, tooth root region reconstruction and tooth alignment, etc. Table 5 shows mean errors of boundary-aware algorithm is smaller than that in uniform simplification method.

TABLE 4
Experiments of boundary-aware and uniform simplification.

Training Data	U(200)		U(1000)	
Simplification Method	Boundary -aware	Uniform	Boundary -aware	Uniform
TGCNNs	**98.55**%	98.37%	**98.93**%	98.62%
TTCNNs	**95.24**%	95.03%	**97.50**%	97.25%
Final	**98.61**%	98.11%	**99.06**%	98.81%

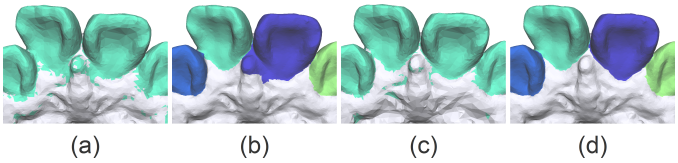


Fig. 7. Comparing boundary-aware and uniform simplification. (a) Teeth-gingiva prediction produced by CNNs on uniformly simplified models. (b) The final results of (a). (c) Teeth-gingiva prediction produced on boundary-aware simplified models. (d) The final results of (c).

TABLE 5
Mean errors of boundary-aware and uniform simplification method.

Training Data	U(200)		U(1000)	
Simplification Method	Boundary -aware	Uniform	Boundary -aware	Uniform
Mean errors/mm	**0.0939**	0.0951	**0.0848**	0.0867

Verification of the effectiveness of label optimization. We employ label optimization in two stages. One is after the GTCNNs to smooth the labeling results for the next stage and the other is after the TTCNNs to again smooth the labeling results. Table 6 shows the effectiveness of label optimization.

TABLE 6
Verification of label optimization.

		CNN Prediction	Label Optimization
TGCNNs	U(1000)	98.93%	99.43%
	L(1000)	98.88%	99.43%
TTCNNs	U(1000)	97.50%	98.56%
	L(1000)	97.37%	98.43%

TABLE 3
Labeling accuracy of different network variants.

	Guo et al. [5]	[5] + COORD	WLF	NN	LeNet	GR-DNN-600	GR-DNN-3	**Ours**
U(1000)	84.81%	95.32%	98.81%	98.35%	98.51%	97.04%	97.00%	**99.06%**
L(1000)	82.95%	95.16%	98.26%	98.04%	98.23%	96.60%	93.42%	**98.79%**

Verification of the effectiveness of improved fuzzy refinement. For general mesh segmentation, label optimization is usually the last step in the pipeline. For normal cases, it is effective and efficient. Dental meshes, as a group of natural models, may have different boundary distribution from the theoretical optimal. The proposed boundary smoothing algorithm slightly adjusts the boundary to make the result much closer to ground truth. The evaluation criteria here is boundary mean errors, i.e., the average error of DCD($S_p \Rightarrow S_{gt}$) and DCD($S_{gt} \Rightarrow S_p$), in which S_p is the computed boundary and S_{gt} is the ground-truth boundary. Table 7 shows its effectiveness.

TABLE 7
Directional Cut Discrepancy (DCD) before and after improved fuzzy refinement.

	DCD($S_p \Rightarrow S_{gt}$)/mm		DCD($S_{gt} \Rightarrow S_p$)/mm	
Tooth Model	Before	After	Before	After
U(1000)	0.0935	**0.0842**	0.0960	**0.0831**
L(1000)	0.0917	**0.0849**	0.0947	**0.0871**

Compared with traditional methods [1], [3], we get better boundary mean errors, shown in Table 8. Table 9 shows the distribution of our boundary mean errors. Besides, their method either requires user interaction or is sensitive to curvatures while our method does not.

TABLE 8
Comparison of boundary mean errors.

Method	Wu et al. [1]	Zou et al. [3]	Ours
Mean errors/mm	0.1218	0.1300	**0.0848**

TABLE 9
Boundary error distribution.

Range/mm		[0,0.25)	[0.25,0.5)	[0.5,∞)
U(1000)	DCD($S_p \Rightarrow S_{gt}$)	92.95%	4.78%	2.27%
	DCD($S_{gt} \Rightarrow S_p$)	93.74%	4.80%	1.46%
L(1000)	DCD($S_p \Rightarrow S_{gt}$)	93.04%	4.69%	2.27%
	DCD($S_{gt} \Rightarrow S_p$)	93.57%	4.66%	1.77%

Fig.8 visualizes the improved boundary by the additional improved fuzzy algorithm. It shows that the wrongly predicted triangle area between two neighbouring teeth is corrected after the refinement. Fuzzy refinement is also able to make up the incomplete part on tooth surface. Such mistakes are commonly seen in results produced by label optimization, thus the improved fuzzy algorithm is indispensable.

Performance with increasing training data. For most learning-based algorithms, the quantity and quality of training data play an important role in the learning step. Although the dental mesh dataset contains more than 1,000

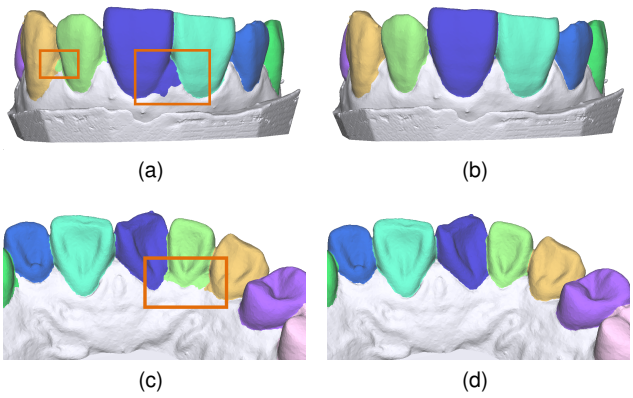


Fig. 8. Comparing results before and after improved fuzzy refinement. Each row represents a dental model. (a)(c) are the results before improved fuzzy refinement, i.e., the results right after label optimization. (b)(d) are fuzzy refinement results.

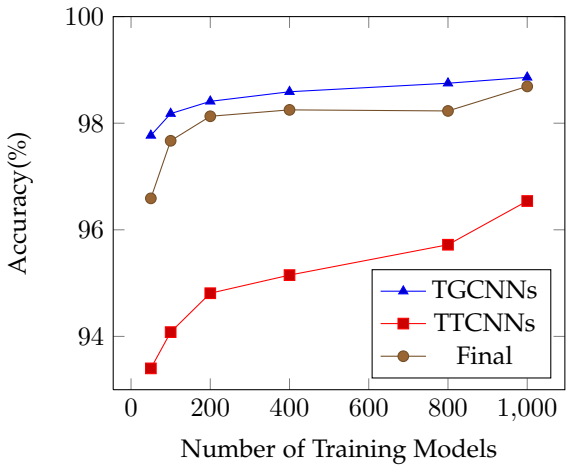


Fig. 9. Experiments with increasing training data on 0.1 simplification ratio.

models, it is necessary to explore how many models are exactly enough to train a good model. According to Fig.9, 1,000 is enough for teeth-gingiva segmentation as the polyline in blue increases very slowly within interval $[400, 1000]$. As for inter-teeth segmentation, the accuracy is still growing in the front half of the red polyline. Suppose that we augment the training data, inter-teeth segmentation and the final results (in brown) may grow for further.

Performance with increasing simplification ratio. The original dental mesh contains too many triangle faces with tiny size, which makes feature extraction time-wasting. On the other hand, an excessively simplified mesh lacks detailed information, so that the extracted features may be less representative. We seek for the relationship between segmentation accuracy on different simplification ratio, shown

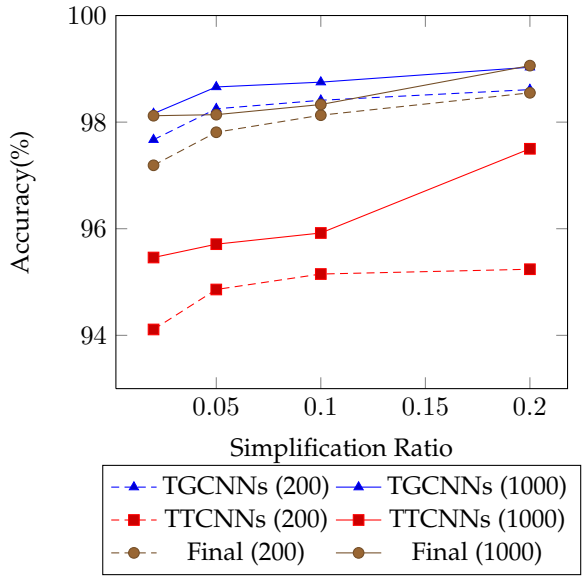


Fig. 10. Experiments with increasing simplification ratio on 200(in dashed polyline) and 1000(in polyline). The color blue, red and brown represent teeth-gingiva segmentation, inter-teeth segmentation and the final results respectively.

in Fig.10. It shows that when the simplification ratio becomes lower, the accuracy decreases. When simplification ratio decreases to 0.02, the absolute face number is too small (4,000 triangles) to retain sufficient features around boundary. On this condition, boundary-aware simplification is not helpful. We suggest a feasible larger ratio value, for example, 0.2, to guarantee enough features for the network to learn.

6 Limitations

The proposed dental mesh segmentation method encompasses a few limitations. The first limitation is that when the boundary between two teeth are corrupted by the simplification process, it will lead to an inaccurate prediction which would be hard to fix even with the improved fuzzy refinement. Thus we encourage a larger simplification ratio. Second, the varying appearances of wisdom teeth on each dental mesh bring in a large portion of the inaccuracy. Some models keep two wisdom teeth there, while some just hold one, and others lack both of them. Some wisdom teeth have erupted completely, while others just show up partly. These make the training step difficult as the amount of wisdom teeth data is very small. As a result, the trained network may tend to mislabel it as gingiva. Another shortage is the 2-level hierarchical network structure. Such structure is suitable for imbalanced dental meshes, while also possesses the common deficiency of decision tree models. In other words, the higher level results will have a negative effect on the lower level. Taking a wisdom tooth as an example, if the teeth-gingiva classification regards it as gingiva, it will no longer take part in the inter-teeth classification step. Under this circumstance, the following label optimization and fuzzy refinement will not help.

7 Conclusion

In this paper, we propose a learning-based dental mesh segmentation method. It receives a detailed 3D dental model as input, and outputs the label list of each triangle face. Different from previous mesh segmentation methods, we propose a label-free mesh simplification method particularly tailored for preserving teeth boundary information while greatly improving the efficiency of the networks. Moreover, we design a hierarchical classification structure based on two CNNs. Both of them are trained on $1,000$ dental meshes. They are not only robust but also generalized well on new models. Last but not least, an improved fuzzy clustering boundary refinement algorithm is raised for the final boundary adjustment. This simultaneous and robust dental mesh segmentation and labeling framework significantly advances the current state-of-the-art geometry-based teeth segmentation methods and achieves 99.06% accuracy for upper dental model, and 98.79% for lower dental model. It directly satisfies the industrial clinical treatment demands and is also robust to any possible foreign matters on dental model surface, e.g., air bubbles, dental accessories, and many more.

Acknowledgments

We thank reviewers for their insightful comments. We are also grateful to all friends for proofreading. This work was supported in part by The National Natural Science Foundation of China No. 61502306, and the China Young 1000 Talents Program.

Appendix

To separate sticky teeth, we solve the following optimization problem.

$$\arg\min_{\{l_i, i\in\mathcal{F}\}} \sum_{i\in\mathcal{F}} \xi_U(p_i, l_i) + \lambda \sum_{i\in\mathcal{F}, j\in\mathcal{N}_i} \xi_S(p_i, p_j, l_i, l_j) \quad (13)$$

The first term is defined as

$$\xi_U(p_i, l_i) = \begin{cases} -\log(p_i) & , \quad l_i \text{ is tooth 1} \\ -\log(1-p_i) & , \quad l_i \text{ is tooth 2} \end{cases} \quad (14)$$

$$\begin{aligned} p_i &= \gamma(P_{\max} - P_{\min}) + 0.5 \\ \gamma &= \max(\min(\frac{(c_i - C)\cdot \hat{d}}{|L|}, P_{\max}), P_{\min}) \end{aligned} \quad (15)$$

where c_i is the barycenter of face i, and C is the barycenter of the sticky-teeth. Constants are $P_{\min} = 10^{-8}$, $P_{\max} = 1 - P_{\min}$, $\lambda = 50$. $\hat{d}$ and $|L|$ are the longest axis's direction and length respectively. The second term is identical to Eqn. 9.

To further smooth the boundary, we apply a shortest-path algorithm within a ring-like area (Fig.13), denoted as $\mathcal{R}$, containing the current boundary, to further smooth the final optimal boundary. For edge v_1v_2, $v_1, v_2 \in \mathcal{R}$, we denote the two faces that share v_1v_2 as i, j. The weight of edge v_1v_2 is

$$\begin{aligned} w_{v_1v_2} &= \|v_1 - v_2\|_2 + \lambda w(\alpha_{ij}) \\ w(\alpha_{ij}) &= \begin{cases} \eta\dfrac{1+\cos\alpha_{ij}}{2}, & \alpha_{ij} < \tau_2 \\ \dfrac{1+\cos\alpha_{ij}}{2}, & \text{else} \end{cases} \end{aligned} \quad (16)$$

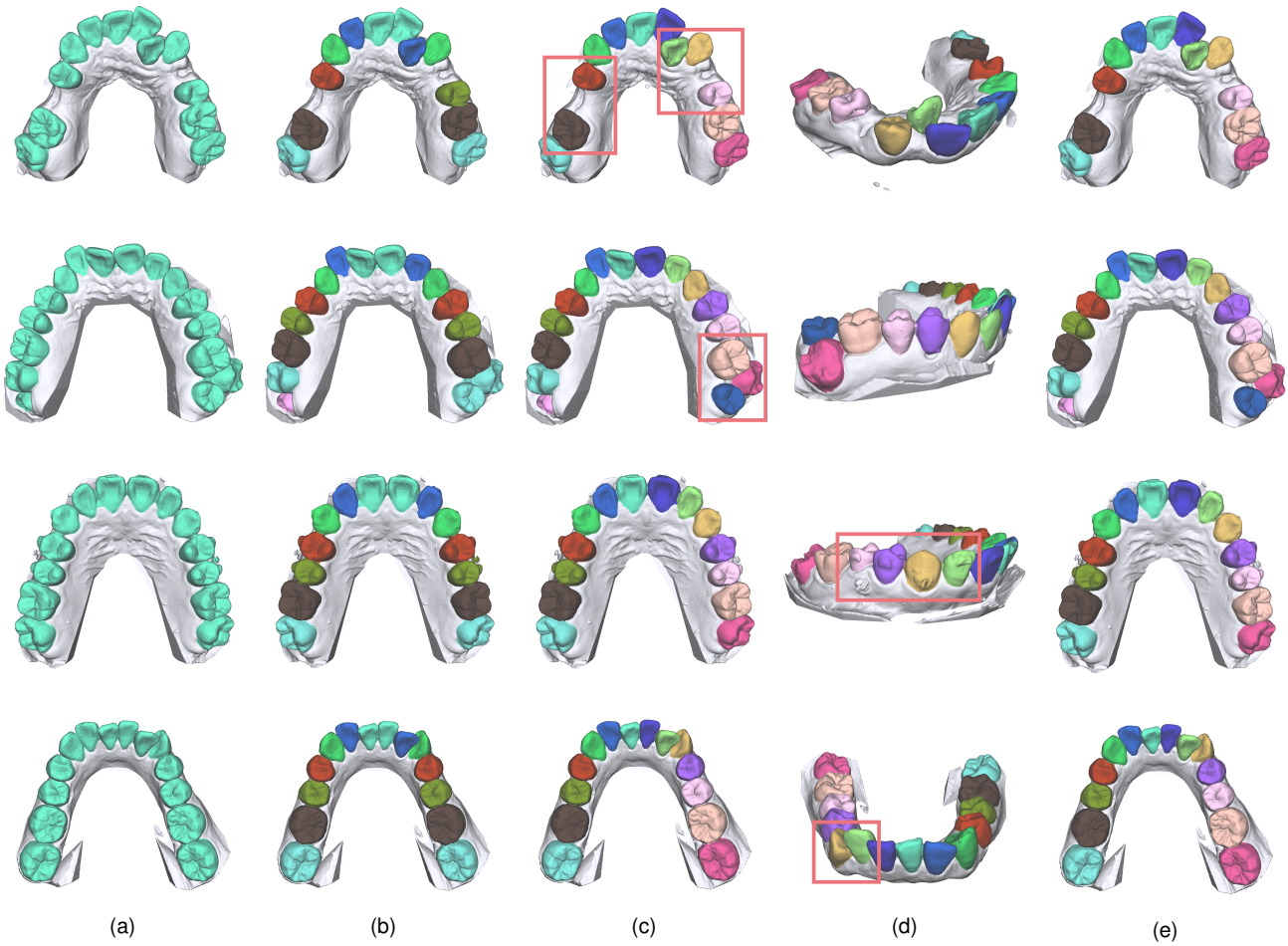


Fig. 11. (a) Teeth-gingiva classification. (b) Inter-teeth classification. (c) Our results. (d) Another view of our results. (e) Ground truth.

where α_{ij} is the same as Eqn. 11. Constants are $\lambda = 0.02, \tau_2 = \frac{\pi}{6}, \eta = 4$. The equation agrees with the fact that edges at groove are more likely to be the boundary. We solve the problem with dynamic programming. Although there are algorithms particularly designed for boundary curve refinement [80], they are complicated to be implemented and do not pay attention to any prior knowledge such as the prediction results in our cases. Our simpler strategy works well in our experiments.

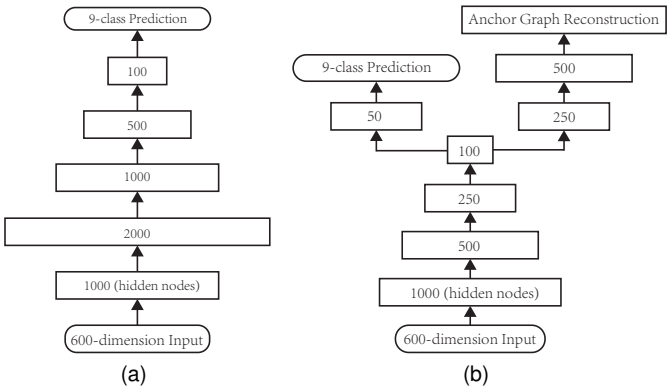


Fig. 12. (a) Neural network structure. (b) GR-DNN structure.

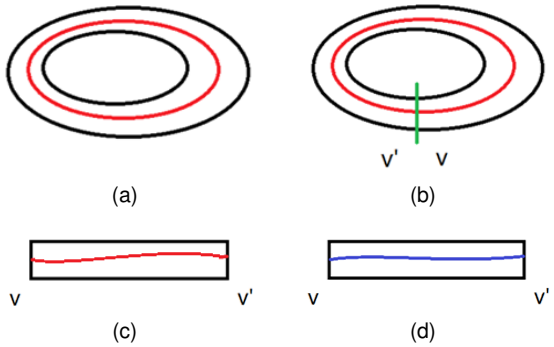


Fig. 13. (a) The black curves represent a narrow ring. The red curve is the initial route. (b) Find a short cut V of the ring and copy the nodes into cut V'. (c) Expand the ring from the cut and get a strap. (d) Find the shortest route from V to V'(the blue curve).

REFERENCES

[1] K. Wu, L. Chen, J. Li, and Y. Zhou, "Tooth segmentation on dental meshes using morphologic skeleton," *Computers and Graphics*, vol. 38, pp. 199–211, 2014.

[2] S. M. Yamany and A. R. Elbialy, "Efficient free-form surface representation with application in orthodontics," *Proceedings of SPIE*, vol. 3640, no. 1, pp. 115–124, 1999.

[3] B. J. Zou, S. J. Liu, S. H. Liao, X. Ding, and Y. Liang, "Interactive tooth partition of dental mesh base on tooth-target harmonic

field," *Computers in Biology and Medicine*, vol. 56, no. C, pp. 132–144, 2015.

[4] M. Zhao, L. Ma, W. Tan, and D. Nie, "Interactive tooth segmentation of dental models," in *Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the*. IEEE, 2006, pp. 654–657.

[5] K. Guo, D. Zou, and X. Chen, "3d mesh labeling via deep convolutional neural networks," *Acm Transactions on Graphics*, vol. 35, no. 1, pp. 1–12, 2015.

[6] M. Attene, S. Katz, M. Mortara, G. Patane, M. Spagnuolo, and A. Tal, "Mesh segmentation - a comparative study," in *IEEE International Conference on Shape Modeling and Applications*, 2006, pp. 7–7.

[7] A. Shamir, "A survey on mesh segmentation techniques," *Computer Graphics Forum*, vol. 27, no. 6, pp. 1539–1556, 2008.

[8] S. Shlafman, A. Tal, and S. Katz, "Metamorphosis of polyhedral surfaces using decomposition," *Computer Graphics Forum*, vol. 21, no. 3, pp. 219–228, 2002.

[9] G. Lavoue, F. Dupont, and A. Baskurt, "A new cad mesh segmentation method, based on curvature tensor analysis," *Computer-aided Design*, vol. 37, no. 10, pp. 975–987, 2005.

[10] S. Katz and A. Tal, *Hierarchical mesh decomposition using fuzzy clustering and cuts*. ACM, 2003, vol. 22, no. 3.

[11] M. Attene, B. Falcidieno, and M. Spagnuolo, "Hierarchical mesh segmentation based on fitting primitives," *The Visual Computer*, vol. 22, no. 3, pp. 181–193, 2006.

[12] A. P. Mangan and R. T. Whitaker, "Partitioning 3d surface meshes using watershed segmentation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 5, no. 4, pp. 308–321, 1999.

[13] Y. Lai, S. Hu, R. R. Martin, and P. L. Rosin, "Fast mesh segmentation using random walks," *Statistical Methods and Applications*, pp. 183–191, 2008.

[14] A. Koschan, "Perception-based 3d triangle mesh segmentation using fast marching watersheds," in *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, vol. 2. IEEE, 2003, pp. II–II.

[15] A. Golovinskiy and T. Funkhouser, "Randomized cuts for 3d mesh analysis," *international conference on computer graphics and interactive techniques*, vol. 27, no. 5, p. 145, 2008.

[16] S. Katz, G. Leifman, and A. Tal, "Mesh segmentation using feature point and core extraction," *The Visual Computer*, vol. 21, no. 8, pp. 649–658, 2005.

[17] L. Shapira, A. Shamir, and D. Cohenor, "Consistent mesh partitioning and skeletonisation using the shape diameter function," *The Visual Computer*, vol. 24, no. 4, pp. 249–259, 2008.

[18] Y. Lee, S. Lee, A. Shamir, D. Cohen-Or, and H. P. Seidel, "Intelligent mesh scissoring using 3d snakes," in *Computer Graphics and Applications, 2004. PG 2004. Proceedings. Pacific Conference on*, 2004, pp. 279–287.

[19] Y. Lee, S. Lee, A. Shamir, D. Cohenor, and H. Seidel, "Mesh scissoring with minima rule and part salience," *Computer Aided Geometric Design*, vol. 22, no. 5, pp. 444–465, 2005.

[20] L. Fan, M. Meng, and L. Liu, "Sketch-based mesh cutting," *Graphical Models graphical Models and Image Processing computer Vision, Graphics, and Image Processing*, vol. 74, no. 6, pp. 292–301, 2012.

[21] M. Meng, L. Fan, and L. Liu, "A comparative evaluation of foreground/background sketch-based mesh segmentation algorithms," *Computers and Graphics*, vol. 35, no. 3, pp. 650–660, 2011.

[22] Z. Ji, L. Liu, Z. Chen, and G. Wang, "Easy mesh cutting," *Computer Graphics Forum*, vol. 25, no. 3, pp. 283–291, 2006.

[23] L. Fan, L. Liu, and K. Liu, "Paint mesh cutting," *Computer Graphics Forum*, vol. 30, no. 2, pp. 603–612, 2011.

[24] Y. Zheng, C. Tai, and O. K. Au, "Dot scissor: A single-click interface for mesh segmentation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 8, pp. 1304–1312, 2012.

[25] M. Meng, L. Fan, and L. Liu, "icutter: a direct cutout tool for 3d shapes," *Computer Animation and Virtual Worlds*, vol. 22, no. 4, pp. 335–342, 2011.

[26] Y. Zheng and C. Tai, "Mesh decomposition with crossboundary brushes," *Computer Graphics Forum*, vol. 29, no. 2, pp. 527–535, 2010.

[27] O. K. Au, Y. Zheng, M. Chen, P. Xu, and C. Tai, "Mesh segmentation with concavity-aware fields," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 7, pp. 1125–1134, 2012.

[28] X. Chen, A. Golovinskiy, and T. Funkhouser, "A benchmark for 3d mesh segmentation," *international conference on computer graphics and interactive techniques*, vol. 28, no. 3, p. 73, 2009.

[29] E. Kalogerakis, A. Hertzmann, and K. Singh, "Learning 3D Mesh Segmentation and Labeling," *ACM Transactions on Graphics*, vol. 29, no. 3, 2010.

[30] Y. Wang, S. Asafi, O. Van Kaick, H. Zhang, D. Cohenor, and B. Chen, "Active co-analysis of a set of shapes," *ACM Transactions on Graphics*, vol. 31, no. 6, p. 165, 2012.

[31] H. Benhabiles, G. Lavoue, J. Vandeborre, and M. Daoudi, "Learning boundary edges for 3dmesh segmentation," *Computer Graphics Forum*, vol. 30, no. 8, pp. 2170–2182, 2011.

[32] T. Kondo, S. H. Ong, and K. W. C. Foong, "Tooth segmentation of dental study models using range images," *IEEE Transactions on Medical Imaging*, vol. 23, no. 3, pp. 350–362, 2004.

[33] M. Grzegorzek, M. Trierscheid, D. Papoutsis, and D. Paulus, "A multi-stage approach for 3d teeth segmentation from dentition surfaces." in *ICISP*. Springer, 2010, pp. 521–530.

[34] N. Wongwaen and C. Sinthanayothin, "Computerized algorithm for 3d teeth segmentation," in *International Conference on Electronics and Information Engineering*, 2010, pp. V1–277–V1–280.

[35] T. Yuan, W. Liao, N. Dai, X. Cheng, and Q. Yu, "Single-tooth modeling for 3d dental model," *International Journal of Biomedical Imaging*, vol. 2010, no. 2010, pp. 1029–1034, 2010.

[36] Y. Kumar, R. Janardan, B. E. Larson, and J. Moon, "Improved segmentation of teeth in dental models," *Computer-aided Design and Applications*, vol. 8, no. 2, pp. 211–224, 2013.

[37] T. Kronfeld, D. Brunner, and G. Brunnett, "Snake-based segmentation of teeth from virtual dental casts," *Computer-aided Design and Applications*, vol. 7, no. 2, pp. 221–233, 2013.

[38] Z. Li, X. Ning, and Z. Wang, "A fast segmentation method for stl teeth model," in *Complex Medical Engineering, 2007. CME 2007. IEEE/ICME International Conference on*. IEEE, 2007, pp. 163–166.

[39] C. Sinthanayothin and W. Tharanont, "Orthodontics treatment simulation by teeth segmentation and setup," in *International Conference on Electrical Engineering/electronics, Computer, Telecommunications and Information Technology, 2008. Ecti-Con*, 2008, pp. 81–84.

[40] Y. Ma and Z. Li, "Computer aided orthodontics treatment by virtual segmentation and adjustment," in *International Conference on Image Analysis and Signal Processing*, 2010, pp. 336–339.

[41] K. Xu, V. G. Kim, Q. Huang, and E. Kalogerakis, "Datadriven shape analysis and processing," *Computer Graphics Forum*, vol. 36, no. 1, 2017.

[42] Z. Barutcuoglu and C. Decoro, "Hierarchical shape classification using bayesian aggregation," in *IEEE International Conference on Shape Modeling and Applications*, 2006, pp. 44–44.

[43] J. W. H. Tangelder and R. C. Veltkamp, "A survey of content based 3d shape retrieval methods," *Multimedia Tools and Applications*, vol. 39, no. 3, p. 441, 2008.

[44] R. Litman, A. Bronstein, M. Bronstein, and U. Castellani, "Supervised learning of bagoffeatures shape descriptors using sparse coding," *Computer Graphics Forum*, vol. 33, no. 5, pp. 127–136, 2014.

[45] Q. Huang, F. Wang, and L. Guibas, "Functional map networks for analyzing and exploring large shape collections," *Acm Transactions on Graphics*, vol. 33, no. 4, pp. 1–11, 2014.

[46] O. Van Kaick, H. Zhang, G. Hamarneh, and D. CohenOr, "A survey on shape correspondence," *Computer Graphics Forum*, vol. 30, no. 6, pp. 1681–1707, 2011.

[47] M. Ovsjanikov, M. Ben-Chen, J. Solomon, A. Butscher, and L. Guibas, "Functional maps: a flexible representation of maps between shapes," *Acm Transactions on Graphics*, vol. 31, no. 4, p. 30, 2012.

[48] X. Guo, J. Lin, K. Xu, and X. Jin, "Creature grammar for creative modeling of 3d monsters," *Graphical Models*, vol. 76, no. 5, pp. 376–389, 2014.

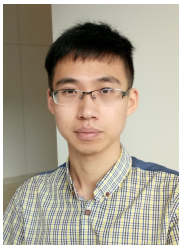
[49] C. Cao, Q. Hou, and K. Zhou, "Displaced dynamic expression regression for real-time facial tracking and animation," *ACM Transactions on graphics (TOG)*, vol. 33, no. 4, p. 43, 2014.

[50] C. H. Shen, H. Fu, K. Chen, and S. M. Hu, "Structure recovery by part assembly," *Acm Transactions on Graphics*, vol. 31, no. 6, pp. 1–11, 2012.

[51] X. Xie, K. Xu, N. J. Mitra, D. Cohen-Or, and B. Chen, "Sketch-to-design: Context-based part assembly," in *Computer Graphics Forum*, 2013, p. 233245.

[52] S. Chaudhuri, E. Kalogerakis, S. Giguere, and T. Funkhouser, "Attribit: content creation with semantic attributes," in *Proceedings of the 26th annual ACM symposium on User interface software and technology*. ACM, 2013, pp. 193–202.

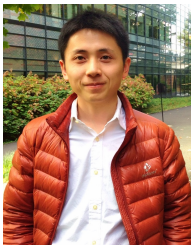
[53] L. Fan, R. Wang, L. Xu, J. Deng, and L. Liu, "Modeling by drawing with shadow guidance," *Computer Graphics Forum*, vol. 32, no. 7, pp. 157–166, 2013.
[54] Y. Wang, M. Gong, T. Wang, D. Cohen-Or, H. Zhang, and B. Chen, "Projective analysis for 3d shape segmentation," *Acm Transactions on Graphics*, vol. 32, no. 6, p. 192, 2013.
[55] W. Xu, Z. Shi, M. Xu, K. Zhou, J. Wang, B. Zhou, J. Wang, and Z. Yuan, "Transductive 3d shape segmentation using sparse reconstruction," *Computer Graphics Forum*, vol. 33, no. 5, pp. 107–115, 2014.
[56] Z. Xie, K. Xu, L. Liu, and Y. Xiong, "3d shape segmentation and labeling via extreme learning machine," *Computer Graphics Forum*, vol. 33, no. 5, pp. 85–95, 2014.
[57] J. Lv, X. Chen, J. Huang, and H. Bao, "Semi-supervised mesh segmentation and labeling," *Computer Graphics Forum*, vol. 31, no. 7pt2, pp. 2241–2248, 2012.
[58] V. G. Kim, W. Li, N. J. Mitra, S. Chaudhuri, S. Diverdi, and T. Funkhouser, "Learning part-based templates from large collections of 3d shapes," *Acm Transactions on Graphics*, vol. 32, no. 4, p. 70, 2013.
[59] O. Sidi, O. Van Kaick, Y. Kleiman, H. Zhang, and D. Cohen-Or, "Unsupervised co-segmentation of a set of shapes via descriptor-space spectral clustering," in *SIGGRAPH Asia Conference*, 2011, p. 126.
[60] R. Hu, L. Fan, and L. Liu, "Cosegmentation of 3d shapes via subspace clustering," *Computer Graphics Forum*, vol. 31, no. 5, pp. 1703–1713, 2012.
[61] P. Shilane, P. Min, M. Kazhdan, and T. Funkhouser, "The princeton shape benchmark," in *Shape Modeling International*, 2004, pp. 167–178.
[62] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, "Real-time human pose recognition in parts from single depth images," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 1297–1304.
[63] K. Xu, V. G. Kim, Q. Huang, and E. Kalogerakis, "Datadriven shape analysis and processing," in *SIGGRAPH Asia*, 2015, p. 4.
[64] Y. Bengio *et al.*, "Learning deep architectures for ai," *Foundations and trends® in Machine Learning*, vol. 2, no. 1, pp. 1–127, 2009.
[65] G. Hinton, "A practical guide to training restricted boltzmann machines," *Momentum*, vol. 9, no. 1, p. 926, 2010.
[66] C. Farabet, C. Couprie, L. Najman, and Y. Lecun, "Learning hierarchical features for scene labeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1915–29, 2012.
[67] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *International Conference on Neural Information Processing Systems*, 2012, pp. 1097–1105.
[68] H. Maron, M. Galun, N. Aigerman, M. Trope, N. Dym, E. Yumer, V. G. Kim, and Y. Lipman, "Convolutional neural networks on surfaces via seamless toric covers," *SIGGRAPH*, 2017.
[69] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 640–651, 2014.
[70] P.-S. Wang, Y. Liu, Y.-X. Guo, S. Chun-Yu, and X. Tong, "O-cnn: Octree-based convolutional neural networks for 3d shape analysis," *ACM Transactions on Graphics (SIGGRAPH)*, vol. 36, no. 4, 2017.
[71] H. Hoppe, "Progressive meshes," in *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*. ACM, 1996, pp. 99–108.
[72] G. Ran and D. Cohen-Or, "Salient geometric features for partial shape matching and similarity," *Acm Transactions on Graphics*, vol. 25, no. 1, pp. 130–150, 2006.
[73] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts. ieee trans pami," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, 2002.
[74] L. Shapira, S. Shalom, A. Shamir, D. Cohen-Or, and H. Zhang, "Contextual part analogies in 3d objects," *International Journal of Computer Vision*, vol. 89, no. 2-3, pp. 309–326, 2010.
[75] A. E. Johnson and M. Hebert, "Using spin images for efficient object recognition in cluttered 3d scenes," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 21, no. 5, pp. 433–449, 1999.
[76] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 23, no. 11, pp. 1222–1239, 2001.
[77] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proceedings of the 22nd ACM international conference on Multimedia*. ACM, 2014, pp. 675–678.
[78] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
[79] S. Yang, L. Li, S. Wang, W. Zhang, and Q. Huang, "A graph regularized deep neural network for unsupervised image representation learning," in *Computer Vision and Pattern Recognition*, 2017, pp. 7053–7061.
[80] L. Kaplansky and A. Tal, "Mesh segmentation refinement," in *Computer Graphics Forum*, vol. 28, no. 7. Wiley Online Library, 2009, pp. 1995–2003.



Xiaojie Xu is a postgraduate student at the School of Information Science and Technology(SIST), ShanghaiTech University. He obtained his B.E. from College of Information Science & Electronic Engineering at Zhejiang University. His research interests include geometry segmentation, deep learning, and virtual reality.



Chang Liu is a postgraduate student at the School of Information Science and Technology(SIST), ShanghaiTech University. She obtained her B.E. from the School of Electronic Engineering at Xidian University. Her research interests include 3D object reconstruction, deep learning on image classification.



Youyi Zheng is a Researcher (PI) at the State Key Lab of CAD&CG, College of Computer Science, Zhejiang University. He obtained his PhD from the Department of Computer Science and Engineering at Hong Kong University of Science & Technology, and his M.Sc. and B.Sc. degrees from the Department of Mathematics, Zhejiang University. His research interests include geometric modeling, imaging, and human-computer interaction.