TAD-Net: tooth axis detection network based on rotation transformation encoding

Yeying Fan, Qian Ma, Guangshun Wei, Yuanfeng Zhou*
The School of Software, Shandong University
Jinan, Shandong Province, China
fyy@mail.sdu.edu.cn, maqiansdu@foxmail.com, guangshunwei@gmail.com, yfzhou@sdu.edu.cn

Zhiming Cui
The University of Hong Kong
Hong Kong SAR, China
cuizm.neu.edu@gmail.com

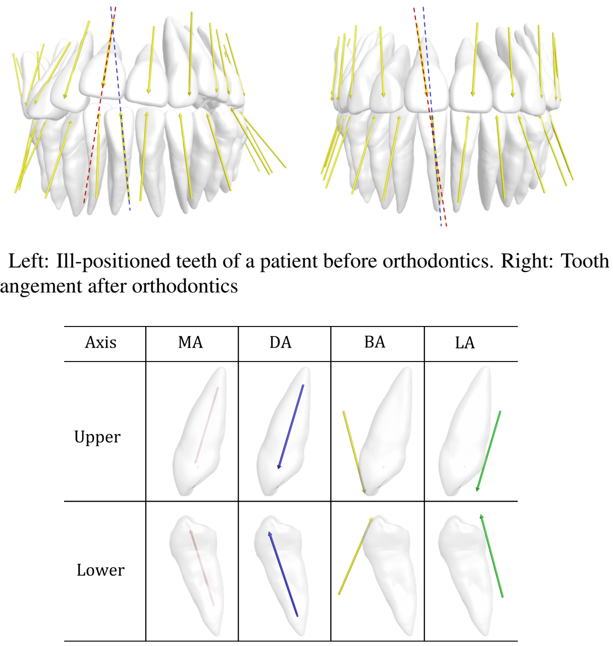
Wenping Wang
Texas A&M University
College Station, Texas, USA
wenping@cs.hku.hk

Abstract

The tooth axes, defined on 3D tooth model, play a key role in digital orthodontics, which is usually used as an important reference in automatic tooth arrangement and anomaly detection. In this paper, we propose an automatic deep learning network (TAD-Net) of tooth axis detection based on rotation transformation encoding. By utilizing quaternion transformation, we convert the geometric rotation transformation of the tooth axes into the feature encoding of the point cloud of 3D tooth models. Furthermore, the feature confidence-aware attention mechanism is adopted to generate dynamic weights for the features of each point to improve the network learning accuracy. Experimental results show that the proposed method has achieved higher detection accuracy on the constructed dental data set compared with the existing networks.

1. Introduction

With the growing concern of oral health and the widely used of computer-aided design (CAD) / computer-aided manufacturing (CAM) in orthodontics [1, 2, 3], digital orthodontics has attracted tremendous attention in recent years. In clinical practice, digital orthodontics enabled by CAD/CAM and data-driven technologies [4] can greatly assist dentists to efficiently make diagnosis or treatment plans. In the workflow of digital orthodontics, dental features including tooth feature axis, points and arch curve defined on 3D tooth models [5], are necessary conditions, as they are usually used as important references in the treatments. In this paper, we focus on the tooth axis detection, which is essential for downstream tasks, such as classification of dental abnormalities [2], tooth redundancy analysis, and tooth arrangement [6].



(a) Left: Ill-positioned teeth of a patient before orthodontics. Right: Tooth arrangement after orthodontics

(b) Tooth axes definition

Figure 1. (a) illustrates the obvious changes of tooth axes before and after orthodontic treatment. (b) illustrates the definition of different tooth axes. The tooth shown here is the left upper canine viewed from the midline position. The blue and red arrows represent distal axis (DA) and mesial axis (MA), and the green and yellow arrows represent buccal surface axis (BA) and lingual axis (LA), respectively.

Specifically, the inclination of a tooth is usually described by the angle between tooth axes as shown in Fig. 1 (a), and the rotation of a tooth is usually described by the

*Corresponding author

movement around the tooth axis. Thus, detecting tooth axis on 3D tooth models is an important problem in clinical applications. Tooth axes, defined by the community of dentistry, are divided into four categories as shown in Fig. 1 (b): Distal Axis (DA), a tangent away from the midline of the tooth and parallel the distal surface; Mesial Axis (MA), a tangent near the midline of the tooth and parallel the mesial surface; Buccal Axis (BA), a tangent is along the inclined surface of tooth near the front of the cheek; Lingual Axis (LA), a tangent near the tongue and parallel to the long axis of the tooth.

In clinical practice, it is laborious and time-consuming for dentists to manually label tooth axes from 3D dental models. Hence, developing an automatic and accurate tooth axis detection method based on 3D tooth models is one of the key problems for digital orthodontics. However, it is still a challenging task due to the following factors. First, these tooth axes are defined by dentists, only considering the functional ability for orthodontic treatments, where there is no clear mathematical definition. Second, the geometric features of dental models are not prominent for accurate tooth axis detection. Thus, it is difficult to detect the tooth axes in a direct regression way which only utilizes the global geometric. Lastly, due to the varying geometric characteristics caused by different types of teeth and tooth axes, designing an unified detection method is extremely challenging.

To address the above challenges, we design a novel learning-based method for automatic tooth axis detection on 3D dental models. The core of our method is to represent the tooth axis in a dense coding rotation transformation way, which is reversible and helpful for the network to learn the context information of the tooth model. Specifically, in our proposed encoding module, the direction field of the tooth model is first introduced to represent the tooth axis densely; then, the direction field is transformed by the corresponding quaternion, and the process is reversible. In this way, the tooth axis detection problem is naturally converted to a dense coding rotation transformation prediction problem. Based upon this encoding method, we designed a rotation transformation prediction network. Moreover, to improve the detection accuracy, especially for different types of teeth and tooth axes with different geometric features, we design a point-wise feature confidence-aware attention mechanism to dynamically learn the point-wise feature weights. Finally, the tooth axis is predicted by the rotation transformation decoding module based on quaternion inverse transformation. Extensive results on benchmarks and ablation studies demonstrate the effectiveness and robustness of the proposed framework.

In summary, we make the following contributions:

- We propose a novel learning-based method for automatic and accurate tooth axis detection, which is the first work in this field to greatly accelerate the workflow of digital orthodontics.
- We convert the tooth axis prediction problem to a point-wise dense rotation transformation prediction task on 3D point cloud. Importantly, a feature confidence-aware attention mechanism is introduced to learn dynamic weights for point-wise features, which in return enhance the learning at reliable points.
- The proposed method can simultaneously predict multiple tooth axes on different kinds of teeth. It has achieved the state-of-the-art performance on the benchmark validated by various experiments.

2. Related Works

2.1. Point cloud learning in dental

3D dental data is the most commonly used data in digital dentistry. And point cloud model is an efficient 3D model representation due to its simple structure and good geometric capture ability. In recent years, the research of point cloud learning has made a breakthrough. PointNet [7] was first proposed. The feature extraction of point clouds is achieved by sampling the features of point clouds. The incoming point cloud is fed into a multilayer perceptron for feature extraction; then, a Max Pooling operation is used to generate representative features. PointNet++ [8] further improved the performance by introducing multi-scale feature extraction scheme to simultaneously encode global and local features. In addition, PointConv [9], DGCNN [10] and other methods can get better results by changing the feature abstraction mode after point cloud sampling and fusing multi-level features as much as possible. After the gradual maturity of point cloud learning technology, many studies have applied it to the 3D tooth model. For example, Ma et al. [11] designed a point cloud model to consider the spatial characteristics of teeth for tooth classification problem. Instead, Zanjani et al. [3] used a deep learning approach for dental segmentation on the intra-oral scanning (IOS) point cloud model. But this kind of square is not fine enough for the segmentation of the boundary and other parts. Cui et al. [12] proposed a two-stage method to segment the tooth point cloud model for accurate style and achieved promising results in the IOS model. These networks achieved state-of-the-art performance on many segmentation and classification tasks, and provided the potential of deep learning in tooth axis detection from 3D tooth models.

2.2. Quaternion-based learning

Quaternion is a kind of hyper-complex number that is widely used in computer graphics and control theory to represent 3D rotation, with the following three main advan-

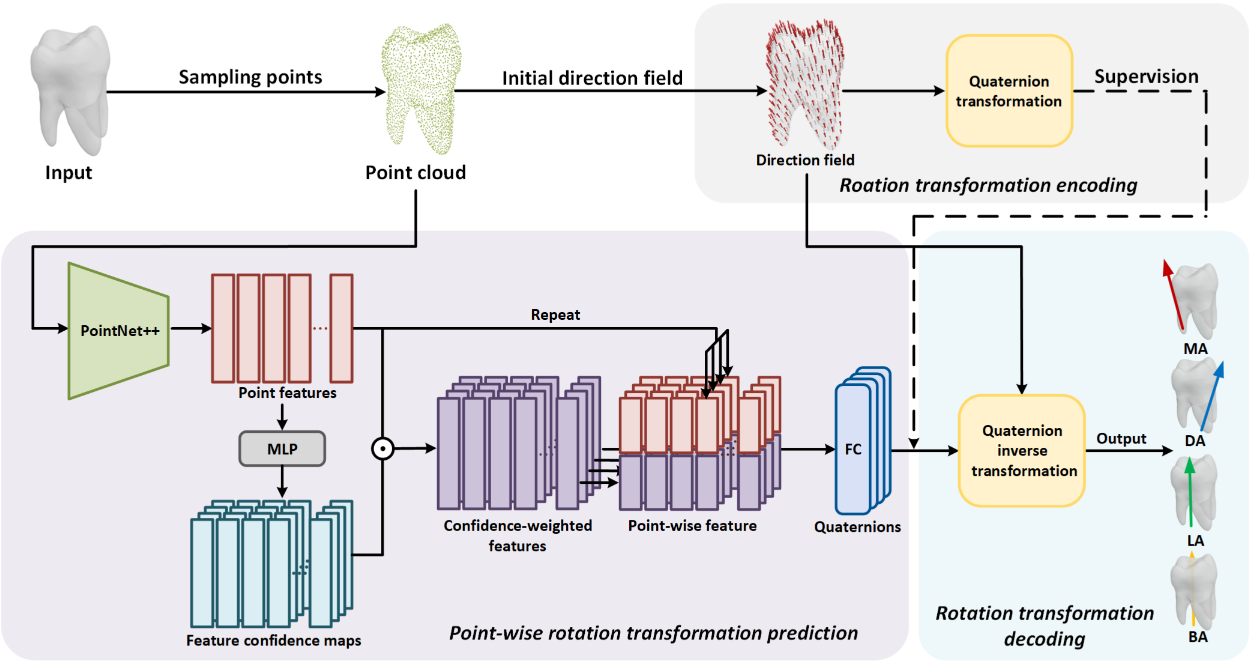

Sampling points
Initial direction field
Input
Point cloud
Direction field
Quaternion transformation
Supervision
Roation transformation encoding
PointNet++
Point features
Repeat
MLP
Feature confidence maps
Confidence-weighted features
Point-wise feature
FC
Quaternions
Point-wise rotation transformation prediction
Quaternion inverse transformation
Output
MA
DA
LA
BA
Rotation transformation decoding

Figure 2. Illustration of the architecture of our proposed tooth axis detection framework. It consists of three components: the rotation transformation encoding module, the point-wise rotation transformation prediction network, and the rotation transformation decoding module. Lastly, we can obtain the final four kinds of tooth axes.

tages: it has solved the Gimbal Lock problem; it only requires storing four parameters, which is much lighter than the rotation matrix; it is more efficient than the rotation matrix in both inverse and series operations. Recently, many works have been explored to introduce quaternion based-learning models [13, 14, 15, 16] to the applications of deep learning. For example, Thomas et al. [17] proposed the *Tensor field networks* (TFN) that are equivariant to 3D rotations, translations, and permutations of localized filters, and it can cover the continuous groups. However, TFN is designed for physics applications with large memory consuming. Based upon the TFN, Zhang et al. [15] proposed a quaternion equivariant capsule network for 3D point clouds to estimate the direction of an object unsupervisedly. Zhao et al. [16] proposed a quaternion product unit to represent data on 3D rotation groups. In addition, Parcollet et al. [13] and [14] introduced a quaternion version to RNN for speech recognition tasks. Different from these existing models, our method combined the quaternion with the dense encoding for tooth axis detection, which detects the tooth axes by utilizing the point-wise predicted rotation transformation way based on the tooth point cloud model.

2.3. Tooth axis detection

In dental clinical applications, tooth orientations are important quantitative metrics in dental diagnosis and surgery planning [1, 2]. Many traditional methods [4, 18, 19, 20] based on the handcrafted geometric features have been proposed to detect tooth axes. Xie et al. [18] detected the tooth axes by using principal component analysis on the segmented soft pulp and dentine, which first detected the individual teeth, and then segmented the soft pulp and dentine for axis prediction. However, the tooth axis detection result of this method is usually affected by the dynamic conditions of neighboring teeth or its surrounding jawbones. Kim et al. [19] proposed an interactive technique to detect axes of single-root teeth. It first interactively segments the tooth, then the tooth axis is obtained by artificially drawing a 2D line on the corresponding slice. The method needs extensive human's interactive adjustments and relies on professional knowledge in orthodontics, which usually affected the detection result greatly. Yang et al. [4] proposed an algorithm to estimate the tooth axes from the 3D CT images for missing teeth. However, the proposed method divided teeth into two classes: single-root teeth and multi-root teeth. Different techniques are proposed to compute tooth axes in different classes. Thus, the proposed method cannot be uniformly applied to different kinds of teeth. In recent years, digital orthodontics have received tremendous research attention. Automation becomes the goal in digital orthodontics [20]. We are the first to explore deep-leaning in the tooth axis detection task, which is an unified method and can be directly applied for all types of teeth with leading performance.

3. Method

In this section, we present a novel framework, named TAD-Net, for tooth axis detection on 3D point cloud models. As shown in Fig. 2, our approach takes the 3D point cloud as input, which is sampled from the input dental model and aims to compute the tooth axes. The rotation transformation from the dense fields of initial direction to the target tooth axes direction field is first encoded by quaternions and view it as the final supervised information (Section 3.1). Then, we introduce the point-wise rotation transformation prediction network and design a feature confidence-aware attention mechanism to improve the tooth axis detection accuracy (Section 3.2). At last, the rotation transformation decoding module is applied to obtain the final tooth axes (Section 3.3).

3.1. Rotation transformation encoding module

In this part, we first initialize a direction vector for each point sampled on a tooth model. In other words, the initialization strategy is to convert the tooth axis into a dense directional field representation. Secondly, the rotation transformation was generated as the supervision information, transformed from the initial direction field into the tooth axes direction field . To realize this rotation transformation, we utilize the quaternion to encode the rotation transformation between initial direction vector and different tooth axes. At last, we will provide the details of the two primary components in the rotation transformation encoding module: direction field representation of tooth model and quaternions-based rotation transformation.

Direction field representation of tooth model. For the direction field (DF), each point in the tooth point cloud model has a corresponding direction vector. Fig. 3 illustrates a direction field that takes a tooth model as an example. The direction field of a tooth point cloud model is a 3D vector field substantially. Thus, the direction field can be defined $DF = \{\mathbf{v}_1, \cdots, \mathbf{v}_n\}$.

Let $D_{train} = \{(T_i, \mathbf{L}_i)\,; i = 1, \cdots, N\}$ is the set of training data, where T_i indicates the tooth point clouds, and $\mathbf{L}_i$ is the corresponding annotated tooth axis. Specifically, for each tooth point cloud T_i, there are three direction fields: $DF_{initial}$, DF_{target} and $DF_{predicted}$. $DF_{initial}$ is the initial direction field, and each point has an initial direction vector $\mathbf{v}$. The initial direction vector is arbitrary and has a slight impact on the final result. Here needs to be explained is that we adopt the unified setting way which is all points have the same initial direction vector for all tooth models in this paper. The related discussions will illustrate in ablation study. DF_{target} is the target direction field (i.e., also named tooth axis direction field). In the DF_{target}, each point direction vector is the tooth axis $\mathbf{L}_i$ annotated on the tooth point cloud T_i. That means the $DF_{target} = \{\mathbf{l}_1, \cdots, \mathbf{l}_n\}$, and $\mathbf{l}_1 = \cdots = \mathbf{l}_n = \mathbf{L}_i$. n is the point number of the single tooth point cloud. $DF_{predicted}$ is the predicted tooth axis direction field or predicted direction field. In the $DF_{predicted}$, each direction vector is the final predicted result. Thus, $DF_{prediction} = \left\{\hat{\mathbf{l}}_1, \cdots, \hat{\mathbf{l}}_n\right\}$, where $\hat{\mathbf{l}}$ is the predicted tooth axis result for a point. Our goal is to find a strategy to accurately predict the rotation transformation from the initial direction vector of $DF_{initial}$ into the target direction vector of DF_{target}. Particularly, the rotation transformation can be learned by a deep learning network. Finally, we utilize the predicted rotation transformation to get the $DF_{prediction}$.

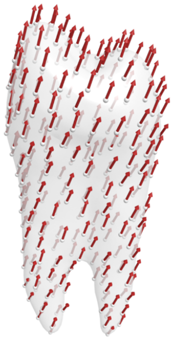


Figure 3. The direction field (DF) of the tooth model. These grey points present the tooth point cloud. Each red arrow presents the direction vector of the point.

Quaternions-based rotation transformation. To our best knowledge, the quaternion is widely used in computer graphics to represent 3D rotation [15], which only requires four parameters to describe a rotation transformation, and it is more efficient than the rotation matrix in both inverse and series operations. Meanwhile, it has a complete system of mathematical theory and can concisely and effectively describe the rotation transformation between the initial direction vector of $DF_{initial}$ and the target direction vector of DF_{target}. Thus, we utilize quaternion algebra to encode the rotation transformation from the initial direction vector to the target direction vector in rotation transformation encoding module, which is named quaternion transformation. Note that it is a reversible transformation. That is, given a initial direction vector of $DF_{initial}$, we can also recover the corresponding tooth axis direction vector of DF_{target} by using quaternion inverse transformation in rotation transformation decoding module.

In this task, quaternion $\mathbf{q}$ in the quaternion domain $\mathbb{H}$, can be reprsented as $\mathbf{q} = s + x\mathbf{i} + y\mathbf{j} + z\mathbf{k}$ and the imaginary units $\mathbf{i}, \mathbf{j}, \mathbf{k}$ obey the quaternion rules that $\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1$, which is a type of hyper complex number with 1D real part s and 3D imaginary part (x, y, z). A quaternion can be represented as the combination of a scalar and a vector as $\mathbf{q} = [s, \mathbf{v}] = [s, (x, y, z)]$ by ignoring the imaginary symbols. Given an initial direction field

$DF_{initial} = \{\mathbf{v}_1, \cdots, \mathbf{v}_n\}$ and a tooth axes direction field $DF_{target} = \{\mathbf{l}_1, \cdots, \mathbf{l}_n\}$, the 3D rotation from a initial direction vector $\mathbf{v}_i \in \mathbb{R}^3$ to the tooth axis $\mathbf{l}_i$ can be described as a rotate around an axis $\mathbf{u}$ with an angle θ:

$$\begin{aligned} &\text{Rotation Axis: } \mathbf{u} = \frac{\mathbf{v} \times \mathbf{l}}{\|\mathbf{v} \times \mathbf{l}\|_2}, \\ &\text{Rotation Angle: } \theta = \arccos\left(\frac{\langle \mathbf{v}, \mathbf{l} \rangle}{\|\mathbf{v}\|_2 \, \|\mathbf{l}\|_2}\right), \end{aligned} \tag{1}$$

where $\mathbf{v} \times \mathbf{l}$ refers to cross product and $\langle \mathbf{v}, \mathbf{l} \rangle$ is for inner product. Each point p is described by a 4-D vector $\mathbf{q} = \left[\cos\left(\frac{1}{2}\theta\right), \sin\left(\frac{1}{2}\theta\right)\mathbf{u}\right]$, where $\|\mathbf{u}\|_2 = 1$. $\|.\|_2$ presents the 2-norm of a vector. We derive a quaternion transformation method to encode the rotation transformation that is concise and complementary as shown in Fig. 4. Finally, the rotation transformation of a tooth model T_i is encoded by quaternions, which can be defined $Q_i = \{\mathbf{q}_1, \cdots, \mathbf{q}_n\}$. Through quaternion encodes the rotation transformation, the rotation transformation has invertibility.

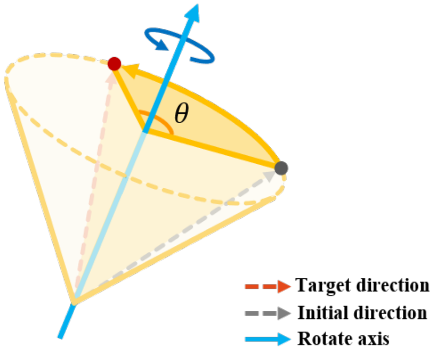


Figure 4. The rotation transformation of a initial direction vector into the target direction vector by a quaternion. The red, gray, and blue arrows represent target direction, initial direction and rotate axis, respectively. θ is the rotate angle. The initial direction vector rotates around an rotate axis with an angle θ to obtain the final direction vector.

3.2. Point-wise rotation transformation prediction network

We design a deep learning network for rotation transformation prediction as shown in Fig. 2. After the rotation transformation encoding module, the point cloud of the 3D dental model is the input of the point-wise rotation transformation prediction network, which mainly includes three components: point-wise feature extraction, feature confidence-aware attention mechanism, and loss function. The details are described as following.

Point-wise feature extraction. Point cloud feature extraction is achieved by sampling and feature aggregation of point clouds. Specifically, the tooth point cloud P sampled from the 3D tooth model is input into the network. Firstly, the farthest point sampling strategy is used to obtain the sampling points from the point cloud, and its neighborhood are chosen within a fixed radius. Then the point cloud features in the neighborhood are aggregated through three layers of multi-layer perceptron (MLP), followed by the batch normalization and ReLU nonlinear activation function. The feature obtained by these operations is the abstraction and fusion of the geometric information of the input point cloud. Finally, the deep feature information, obtained after multiple sampling and feature aggregation for the original point cloud model, can be used for this task.

Feature confidence-aware attention mechanism. Since different tooth axes have various geometric characteristics, the feature distribution of the input tooth point cloud exists significant difference. We should pay attention to those reliable points with stronger feature expression ability so as to highlight the discrepancies among different feature distributions. Based upon this idea, the feature confidence-aware attention mechanism, inspired by [21] [22][23], is introduced to further improve detection accuracy. In this module, the feature confidence-aware attention mechanism measures the reliability of each point by learning a confidence map with dynamic weights for different point features. The confidence-weighted features, using for prediction, were obtained by multiplying each point feature and its corresponding feature confidence map. The higher the confidence value is, the more accurate the prediction result is. However, the confidence-weighted features only focus on the higher confidence value points and ignores these points whose feature confidence value is lower. Thus, we concatenate the extracted all point features and confidence-weighted features together. Then, the final point-wise features were fed into the fully connected layer to predict the point-wise rotation transformation of the tooth model. The mechanism is the same for different tooth axes.

Loss function. Given the point cloud of the tooth model T_i and the ground truth quaternions $Q_i = \{\mathbf{q}_1, \cdots, \mathbf{q}_n\}$, the point-wise transformation prediction network outputs corresponding point-wise quaternion vectors. The overall direction field of tooth model prediction loss function $\mathcal{L}$ is defined as:

$$\begin{aligned} \mathcal{L} &= \frac{1}{N} \sum_{i=0}^{N} \mathcal{L}_i^{\mathrm{q}}, \\ \mathcal{L}_i^{\mathrm{q}} &= \frac{1}{n} \sum_{j=0}^{n} \left(\mathbf{q}_j - \hat{\mathbf{q}}_j\right)^2, \end{aligned} \tag{2}$$

where $\mathcal{L}_i^{\mathrm{q}}$ is the prediction loss of tooth T_i, which indicates the point-wise loss between the predicted quaternion vector $\hat{\mathbf{q}}$ and the ground truth quaternion vector $\mathbf{q}$. We utilize the mean square error (MSE) as loss function to calculate the regression error.

3.3. Rotation transformation decoding module

With the help of quaternion algebra, we propose the rotation transformation decoding module, which introduces how to get the predicted tooth axis $\hat{\mathbf{L}}$ by quaternion inverse transformation from the predicted quaternions $\hat{Q}$ and the initial direction field $DF_{initial}$. For the initial direction vector $\mathbf{v}$ of each point in the $DF_{initial}$, we can use the following equation to obtain the final predicted tooth axes $\hat{\mathbf{l}}$:

$$\left[0, \hat{\mathbf{l}}\right] = \hat{\mathbf{q}} \left[0, \mathbf{v}\right] \hat{\mathbf{q}}^*, \tag{3}$$

where $\hat{\mathbf{q}} = [s, \mathbf{v}]$ is the predicted quaternion and $\hat{\mathbf{q}}^* = [s, -\mathbf{v}]$ stands for the conjugation of $\hat{\mathbf{q}}$. $\left[0, \hat{\mathbf{l}}\right]$ and $[0, \mathbf{v}]$ represent the pure quaternions of $\hat{\mathbf{l}}$ and $\mathbf{v}$.

For each tooth point cloud model T_i, we can get the predicted direction field $DF_{prediction} = \left\{\hat{\mathbf{l}}_1, \ldots, \hat{\mathbf{l}}_n\right\}$. Then, we obtain the final tooth axis $\hat{\mathbf{L}} = \frac{1}{n}\sum_j^n \hat{\mathbf{l}}_j$ by calculating the average of the dense predicted tooth axes $\hat{\mathbf{l}}$. The decoding method is applied for all kinds of tooth axes in the same way. Finally, we can obtain four kinds of tooth axes simultaneously.

4. Experiments

In this section, we present detailed experimental results and analysis of the proposed method, demonstrating the effectiveness of different key modules and the leading performance compared to other methods.

4.1. Implementation and training details

Network details. In this framework, we use the PointNet++ network as our backbone. Given an input tooth model, we first extract the mesh vertices and uniformly sample it to obtain the input point cloud with dimension $n \times 3$, where $n = 2500$ is the number of sampled input points 3D coordinates describe each point. Having the input point cloud, we first set all the vectors of the initial direction field of each tooth point cloud to $[0, 0, 1]$, and use the quaternion obtained by the quaternion transformation module as the supervision information. Then, we normalize the tooth model within a unit ball and extract the point-wise features by three set abstractions and three feature propagations, which include three blocks of multi-layer perceptrons (MLPs) followed by a batch normalization layer and a ReLU nonlinearity layer. The output of the backbone is a set of features of subsampled points and sent to a multi-layer perceptrons layer for automatically learning the point-wise confidence value. The feature confidence map is multiplied with the corresponding feature to obtain the confidence-weighted feature. We concat the confidence-weighted feature with the point-wise feature, which is outputted from the backbone to obtain the final point-wise feature for quaternion prediction. Finally, the tooth axes can be obtained through the quaternion inverse transformation module.

Training details. All the experiments are conducted on a single Nvidia GeForce RTX 3090Ti GPU. Our method is trained using the ADAM optimizer [24]. The learning rate and weight decay rate are set to 1×10^{-3} and 0.9, respectively. The batch size is set to 64 for maximizing the GPU memory occupancy. We train the network on the PyTorch [25] platform and stop the training at 800 epochs as the validation loss no longer decreases. The total training time is about 8 hours.

4.2. Dataset

We trained and validated on a dataset that includes a total of 2910 3D point cloud tooth models. The dataset comes from a medical research institution that has scientific research cooperation with us. To obtain the ground truth, the dataset was manually annotated with the corresponding feature axes by professional dentists. Each tooth axis is independently labeled by four professional dentists, then removing invalid data with the largest variance. Finally, the ground truth is obtained by averaging the valid labeled data. The ground truth is labeled in four classes: Buccal surface axis, Lingual axis, Mesial axis, and Distal axis. To train the neural network, the dataset was randomly split into two subsets, a training set and a testing set with 2034 and 876 3D tooth models, respectively.

4.3. Evaluation metrics

We define two reasonable evaluations for tooth axis to evaluate the detection accuracy in the testing set.

Error angle. The test dataset is defined by $D_{test} = \left\{\left(T_i, \hat{\mathbf{L}}_i\right); i = 1, \cdots, N\right\}$. We use the error angle E_{angle} to evaluate the detection accuracy of tooth axes in test dataset since the tooth axis is an independent coordinate vector.

$$E_{\text{angle}} = \frac{1}{N}\sum_{i=1}^{N} \arccos\left(\hat{\mathbf{L}}_i, \mathbf{L}_i\right), \tag{4}$$

where $\hat{\mathbf{L}}_i$ is the predicted tooth axis of the tooth T_i and $\mathbf{L}_i$ is the corresponding ground truth axis. N is the number of tooth axes in testing dataset.

Angle average precision of tooth axes. To further analyze the results, we proposed a metric named the angle Average Precision (aAP). This metric is inspired by the mean average precision. The predicted tooth axis $\hat{\mathbf{L}}_i$ is considered to be a positive sample or correct if and only if

$$\arccos\left(\hat{\mathbf{L}}_i, \mathbf{L}_i\right) \leq \theta_L, \tag{5}$$

where θ_L is a predefined threshold, and $\mathbf{L}_i$ is the ground truth tooth axis of T_i. The resting will be considered to be negative samples. Specifically, we set the threshold θ_L to 5°, 10°, 15°, denoted by aAP^{5°, aAP^{10°, aAP^{15°.

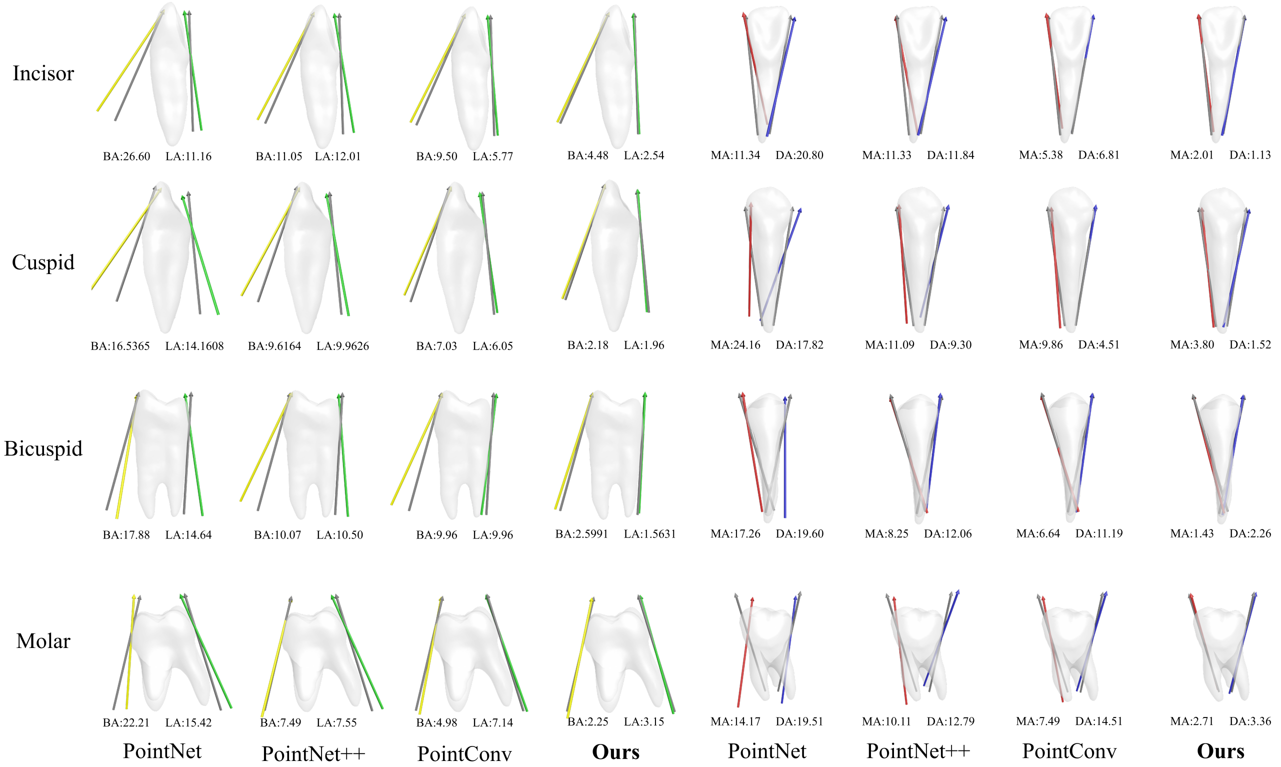


Figure 5. The visual comparison of tooth axis detection results produced different methods. Each row corresponding to a typical example of tooth axes with different color arrow: BA (yellow arrow), LA (green arrow), MA (blue arrow), DA (red arrow). The gray arrow stands for the ground truth result. From left to right are other comparison methods (the 1st-3rd and 5th-7th columns) and the result of our method (the 4th and 8th columns). There are different types of tooth models to show the applicability of our method.

4.4. Results

The last row of Tab. 1 and Tab. 2 show the tooth axis detection accuracy. It can be seen that we have achieved excellent results for all kinds of tooth axes, and the predicted tooth axes match well to the ground truth. It should be noted that the training and testing sets contain multiple types of teeth, so our method can be successfully applied to different kinds of teeth. In addition, Fig. 5 visualizes the tooth axis prediction results from the testing set, which shows the results of different tooth axes on different types of teeth. It can be seen that the predicted tooth axes produced by our method are closed to the ground truth annotated by the dentist for all kinds of teeth and tooth axes.

4.5. Comparisons

In order to verify advantage of our proposed the network (TAD-Net), we compared our method with other performance. However, considering there are few methods designed for tooth axis detection in existing studies, and there is no published dataset, it is difficult to conduct direct comparative experiments. Thus, we use the same point cloud as the input and compare our method with several typical point cloud learning frameworks to directly regress the tooth axes, including PointNet[7], PointNet++[8] and PointConv[9]. These methods take point cloud as input, and the output is four kinds of tooth axes. The overall tooth axis detection results are summarised in Tab. 1 and Tab. 2.

- PointNet: We use the point cloud as input. There are 2500 points in the point cloud. After the input data is processed by T-net, we use three Conv1d layers [64,128,1024] to extract features, and output 1024-dimensional feature vector as the global feature. Then, the global features and the local features of each point are concated into four fully connected layers.

- PointNet++: Compared with PointNet, PointNet++ provides more sampling sizes in point cloud. We also use the same sampling method as our method backbone (i.e., PointNet++) and obtain multi-scale features. Finally, we extract 256 dimensional feature vectors from the point cloud followed by four fully connected layers to predict the final results.

- PointConv: The architecture of PointConv in this article remains similar to that introduced in the original

Table 1. Comparison of different networks for tooth axis detection based on E_{angle} (°).

	MA	DA	LA	BA	Average
PointNet	13.85	15.48	12.93	12.58	13.71
PointNet++	10.89	11.77	10.15	9.36	10.54
PointConv	8.56	11.01	11.28	9.01	9.97
Our method	**2.78**	**3.42**	**3.65**	**3.48**	**3.33**

Table 2. The aAP for different methods.

	Axis	aAP^{5°	aAP^{10°	aAP^{15°
PointNet	BA	5.17	13.22	24.64
	LA	5.29	11.78	22.72
	MA	4.09	10.34	21.63
	DA	3.25	9.13	18.27
PointNet++	BA	7.09	18.27	32.93
	LA	5.65	14.06	26.44
	MA	3.37	9.62	17.67
	DA	3.61	8.29	14.3
PointConv	BA	8.29	24.04	43.39
	LA	9.38	21.27	37.26
	MA	5.29	15.99	31.61
	DA	6.97	16.83	28.73
Ours	BA	42.91	69.47	82.33
	LA	48.08	73.32	84.62
	MA	41.59	65.63	78.49
	DA	40.63	63.94	79.09

paper. To have a fair comparison with other competing methods, we also use 3D coordinate information as the network input without 3D normal information.

We conducted experiments under the same hardware and parameter conditions. From the experimental results in Tab. 2, we can see that our network achieves better results under the same conditions and different threshold values. That shows that the method by directly using dense rotation transformation benefits the tooth axis detection. In addition, the way that learns dynamic weights for point-wise features can make the prediction results perform better. Fig. 5 shows the performance of the four methods on different tooth classes and axes. These figures from Fig. 5 strongly show the advantages of our method in dealing with the tooth axes detection problem.

4.6. Ablation Study

To verify the effectiveness of our framework components, including the rotation transformation encoding module for dense tooth axis representation and the feature confidence-aware attention module in the direction field prediction network. In addition, we also discuss the influence of the setting of the initial vector in the initial direction field. We use error angle E_{angle} as the evaluation metrics.

Table 3. Ablation study of the tooth axis detection. DE means dense encoding module of tooth axes. RT stands for the rotation transformation ecoding based on the dense encoding method. FCA refers to feature confidence-aware attention mechanism. FMC represents point-wise feature and feature confidence map concatenation (°).

	DE	RT	FCA	FMC	MA	DA	LA	BA	Average
(a)					10.89	11.77	10.15	9.36	10.54
(b)	✓				4.70	4.93	5.10	5.69	5.11
(c)	✓	✓			3.83	3.80	3.84	4.46	3.98
(d)	✓		✓		4.27	4.88	4.82	5.30	4.82
(e)	✓		✓	✓	3.60	4.79	4.86	4.46	4.42
(f)	✓	✓	✓	✓	**2.78**	**3.42**	**3.65**	**3.48**	**3.33**

Tab. 3 summarizes the comparisons. We describe the detailed results of ablation studies in the following.

Rotation transformation encoding module of tooth axes. The rotation transformation encoding is a dense encoding way. To validate the importance of the rotation transformation encoding module of tooth axes, we first train a network to detect tooth axes using the direct regression method. The results of the direct regression method are shown in the first row in Tab. 3. The direct regression method actually only utilizes the global geometric features. Therefore, many cases are failed due to missing local geometric information. Then, the dense encoding is introduced into the baseline. Point-wise direction vector is regressed corresponding to the tooth axis. The results are shown in the second row (b). Obviously, dense coding is more suitable for tooth axis detection, as the prediction results are greatly improved. Finally, we also designed the rotation transformation encoding and decoding module. The results are shown in the third row (c). Through these two modules, the tooth axis prediction is transformed into the prediction of directional field transformation. The rotation transformation is obtained through the feature prediction of each point, which greatly improves the feature extraction ability of each point in the point cloud and obtains more accurate prediction results.

Feature confidence-aware attention mechanism module. The feature confidence-aware attention mechanism is a dynamic attention mechanism. This mechanism constantly changes the weight of features in the training process. According to the needs of the task, it assigns different weights to the features in different channels, which can potentially learn the importance of different features and enhance the learning at those points with more accurate prediction results. The results are shown in Tab. 3. By comparing the results of rows (b) and (d), we can see that the network with this mechanism achieves better performance.

Point-wise feature and confidence-weighted feature concatenation. The skipped connections concatenate the point-wise feature and confidence-weighted feature. The

Table 4. Results of different initial direction vector setting (°).

	MA	DA	LA	BA	Average
category1	19.29	18.08	18.04	17.80	18.30
category2	16.44	13.28	15.19	12.25	14.29
category3_1	3.31	3.43	3.22	3.79	3.44
category3_2	2.78	3.42	3.65	3.48	3.33

Table 5. Results of different backbone network (°).

	MA	DA	LA	BA	Average
PointNet	7.75	7.17	8.74	6.96	7.66
PointConv	4.76	4.45	3.16	3.43	3.95
PointNet++	2.78	3.42	3.65	3.48	3.33

confidence-weighted feature is obtained by the original point-wise feature multiplied by the corresponding feature confidence map. The feature confidence attention mechanism strengthens the features with high confidence and suppresses some features with weak expressiveness. To avoid losing weak feature information in learning, we concatenate the confidence-weighted feature with the point-wise feature to increase the weight of some more confident features but simultaneously without losing the original point-wise feature information. The visual results of row (d) and row (e) are consistent with the statistical results presented in the Tab. 3.

Initial direction vector setting. To verify the effect of the initial direction vector setting in our method, we designed three experiments with different initial direction vector fields. Since our method is the point-wise prediction, the first predefined initial direction vector field is that all points sampled on a 3D tooth model have different direction vectors. For example, we take the normal field as the initial direction field. The second category is that each point in the same tooth model has the same direction vector, and different tooth models have different initial direction vectors. To meet this requirement, we calculate the principal direction for each tooth model by principal components analysis (PCA), where the main direction was taken as the initial direction vector of all points in the corresponding tooth model. The third category is that all points have the same initial direction vector for all tooth models. Thus, the initial direction vector can be set to any vector. In the experiment, we randomly select vectors $[2, 1, 4]$ and $[0, 0, 1]$ as category3_1 and category3_2 to prove it. The visual and statistical results are presented in Tab. 4 and Fig. 6, showing that the non-uniform initialization direction field makes the learning of rotation transformation more difficult, and it is very hard to learn a consistent result from disordered initial data. Thus, the unified initialization can improve the performance of the model.

Backbone network. To verify the effectiveness of the dense encoding module based on direction field and the feature confidence-aware attention mechanism module for the tooth axis detection, we conducted an ablation experiment that replaced the backbone network in our method with PointNet, and PointConv [9], respectively. From the results in the Tab. 5, we can see that our backbone network is flexible and can achieve better performance when using PointNet++ as the backbone network.

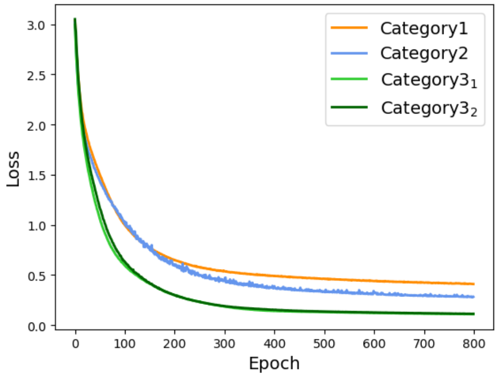


Figure 6. Loss curves of different initial direction vector setting. The abscissa and ordinate represent the number of iterations during the training period and corresponding loss.

5. Discussion

5.1. Applications

Various applications can potentially benefit from tooth axis prediction. In traditional dentistry, dentists measure the tooth axes on the incisor to diagnose the occlusal condition of patients based on prior knowledge, and then perform a series of follow-up treatments. However, it often depends on the professional experience of a dentist. Also, in digital dentistry, the tooth axis is more widely used since digital dentistry can automatically measure more accurate tooth axes for each tooth and analyze it energetically, more information can be obtained through the tooth axes. For example, in the applying of tooth axis, the two of the most important applications are the detection and classification of tooth abnormalities and the tooth arrangement with tooth axes.

Detection and classification of abnormal tooth. This task is to determine whether the teeth are abnormal by comparing the tooth axis distribution with the ideal occlusal state. Normally, in an ideal occlusal state, the tooth axes are evenly distributed along the arch line and symmetrically distributed relative to the occlusal midline. The angle between the same type of adjacent teeth is small and the directions are close. The tooth axis from front to back presents the characteristics of smooth changes, and the angle between the corresponding upper and lower tooth axis

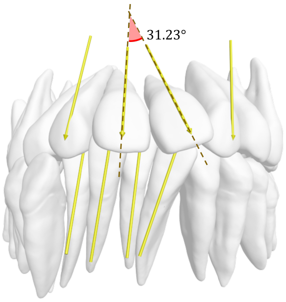
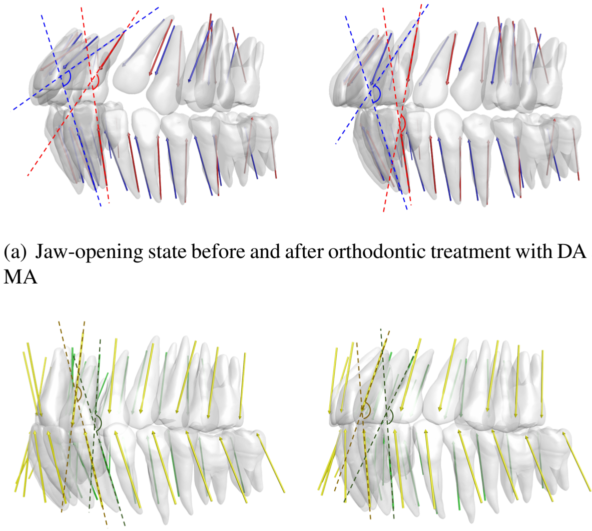


Figure 7. Abnormal tooth: Mesial malocclusion. This angle is formed by the BA axis of the two upper incisors. In an ideal state, the angle formed by the two BA is closed to parallel.

should also be kept within a reasonable angle range. If there is a large difference in the tooth axis distribution compared to the ideal occlusion, the abnormal situation in the tooth can be judged accordingly. For example, if the angle of adjacent teeth is large, there may be teeth crowding, malocclusion problems, as shown in Fig. 7, where the angle of intersection of the BA of the two incisors in the upper teeth reaches 31.23°. In an ideal occlusion, the two BA should be close to parallel, so it can be judged that this case may have mesial malocclusion. Also, if the angle of the teeth axis between the upper and lower teeth of the front teeth is too small, there may be the front teeth protruding, deep overbite jaw and so on.



(a) Jaw-opening state before and after orthodontic treatment with DA and MA

(b) Underbite state before and after orthodontic treatment with BA and LA

Figure 8. Results before and after orthodontic treatment. The left presents a set of ill-positioned teeth arrangements before orthodontic treatment. The right presents the ideal tooth arrangement based on the tooth axes by our method detected.

Tooth arrangement with tooth axes. Tooth arrangement is a crucial and necessary step in orthodontic treatment [6, 26, 27]. This task aims to obtain a ideal tooth arrangement state from a set of ill-positioned teeth of a patient, as shown in Fig. 8. In the traditional dental orthodontic process, dentists need to comprehensively consider the overall occlusal state of patients, formulate treatment plans, and then wear appliances to achieve the purpose of orthodontics. Therefore, in these steps, we heavily rely on the dentist's professional experience to design the orthodontic plan. However, in digital dentistry, the deformity of teeth can be accurately measured by calculating the tooth axis. According to some measurement indicators of malocclusion, the orthodontic scheme of teeth can be made automatically to reduce the difference between the tooth axis of ideal occlusion and the tooth axis of malocclusion.

5.2. Limitations and future work

For tooth axis detection and the related applications, our current method has certain limitations, and we believe it will inspire and promote future works.

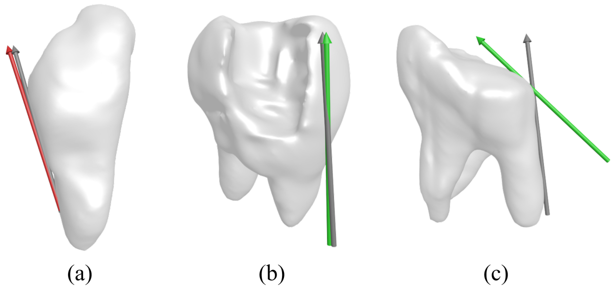


Figure 9. (a) and (b) are slightly incomplete tooth models. (c) is an extremely incomplete tooth model. For (a) and (b), our method can still get better prediction results. However, the prediction result of (c) is not ideal since the shape has changed greatly.

Limitations. Although our method has achieved excellent experimental results on our dataset, it still has some limitations. For example, the tooth point cloud model is extremely abnormal or incomplete, which leads to the tooth geometric information being inaccurate. For slightly incomplete or abnormal, we can still achieve better results. Thus, these problems are still a challenge to our method. There are some predict failure cases shown in Fig. 9. In addition, the dense coding increases the supervision information and improves the detection accuracy while the amount of parameters also increases. In the process of network training, it relatively takes more time for parameter learning and network convergence compared with the direct regression method. In the future, we will explore the method of tooth axis detection with lower resource costs and higher accuracy.

Future work. Our network can only detect the tooth axes. The relevant applications of tooth axes (i.e., Tooth arrangement, detection and classification of abnormal tooth and tooth redundancy analysis) are calculated separately and not included in the network. It would still be interesting to integrate the tooth axis detection and other relevant applications in our an end-to-end pipeline.

6. Conclusion

In this work, we develop a novel automatic tooth axis detection method. The method encodes the rotation transformation of tooth point cloud model, which transforms the tooth axes prediction into the direction field prediction represented by quaternions. Lastly, we decode the predicted direction field to get the final tooth axes. In the network, the confidence-aware attention mechanism is employed to dynamically learn weights for the features of each point for improving detection accuracy. We have evaluated our algorithm both qualitatively and quantitatively, and compared it with the related methods. Extensive experiments prove our method that produces superior results and significantly outperforms others. This method of accurately detecting the tooth axis will be of great significance for digital dentistry.

References

[1] Joanneke M Plooij, Thomas JJ Maal, Piet Haers, Wilfred A Borstlap, Anne Marie Kuijpers-Jagtman, and Stefaan J Bergé. Digital three-dimensional image fusion processes for planning and evaluating orthodontics and orthognathic surgery. a systematic review. *International journal of oral and maxillofacial surgery*, 40(4):341–352, 2011. 1, 3

[2] Christos C Galanis, Michael M Sfantsikopoulos, Petros T Koidis, Nikolaos M Kafantaris, and Pavlos G Mpikos. Computer methods for automating preoperative dental implant planning: Implant positioning and size assignment. *Computer methods and programs in biomedicine*, 86(1):30–38, 2007. 1, 3

[3] Farhad Ghazvinian Zanjani, David Anssari Moin, Bas Verheij, Frank Claessen, Teo Cherici, Tao Tan, et al. Deep learning approach to semantic segmentation in 3d point cloud intra-oral scans of teeth. In *International Conference on Medical Imaging with Deep Learning*, pages 557–571. PMLR, 2019. 1, 2

[4] Yang Wang, Lin Wu, Huayan Guo, Tiantian Qiu, Yuanliang Huang, Bin Lin, and Lisheng Wang. Computation of tooth axes of existent and missing teeth from 3d ct images. *Biomedical Engineering/Biomedizinische Technik*, 60(6):623–632, 2015. 1, 3

[5] Lawrence F Andrews. The six keys to normal occlusion. *Am J orthod*, 62(3):296–309, 1972. 1

[6] Guodong Wei, Zhiming Cui, Yumeng Liu, Nenglun Chen, Runnan Chen, Guiqing Li, and Wenping Wang. Tanet: Towards fully automatic tooth arrangement. In *European Conference on Computer Vision*, pages 481–497. Springer, 2020. 1, 10

[7] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017. 2, 7

[8] Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv preprint arXiv:1706.02413*, 2017. 2, 7

[9] Wenxuan Wu, Zhongang Qi, and Li Fuxin. Pointconv: Deep convolutional networks on 3d point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9621–9630, 2019. 2, 7, 9

[10] Anh Viet Phan, Minh Le Nguyen, Yen Lam Hoang Nguyen, and Lam Thu Bui. Dgcnn: A convolutional neural network over large-scale labeled graphs. *Neural Networks*, 108:533–543, 2018. 2

[11] Qian Ma, Guangshun Wei, Yuanfeng Zhou, Xiao Pan, Shiqing Xin, and Wenping Wang. Srf-net: Spatial relationship feature network for tooth point cloud classification. In *Computer Graphics Forum*, volume 39, pages 267–277. Wiley Online Library, 2020. 2

[12] Zhiming Cui, Changjian Li, Nenglun Chen, Guodong Wei, Runnan Chen, Yuanfeng Zhou, and Wenping Wang. Tsegnet: An efficient and accurate tooth segmentation network on 3d dental model. *Medical Image Analysis*, 69:101949, 2021. 2

[13] Titouan Parcollet, Ying Zhang, Mohamed Morchid, Chiheb Trabelsi, Georges Linarès, Renato De Mori, and Yoshua Bengio. Quaternion convolutional neural networks for end-to-end automatic speech recognition. *arXiv preprint arXiv:1806.07789*, 2018. 3

[14] Titouan Parcollet, Mirco Ravanelli, Mohamed Morchid, Georges Linarès, Chiheb Trabelsi, Renato De Mori, and Yoshua Bengio. Quaternion recurrent neural networks. *arXiv preprint arXiv:1806.04418*, 2018. 3

[15] Xuan Zhang, Shaofei Qin, Yi Xu, and Hongteng Xu. Quaternion product units for deep learning on 3d rotation groups. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7304–7313, 2020. 3, 4

[16] Yongheng Zhao, Tolga Birdal, Jan Eric Lenssen, Emanuele Menegatti, Leonidas Guibas, and Federico Tombari. Quaternion equivariant capsule networks for 3d point clouds. In *European Conference on Computer Vision*, pages 1–19. Springer, 2020. 3

[17] Nathaniel Thomas, Tess Smidt, Steven Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick Riley. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*, 2018. 3

[18] Jie Xie, Yang Liu, Jilin Yin, Tiantian Qiu, Yuanliang Huang, and Lisheng Wang. An interactive method for computing tooth-root orientation from 3d ct image. In *2010 International Conference of Medical Image Analysis and Clinical Application*, pages 42–45, 2010. 3

[19] Gyehyun Kim, Jeongjin Lee, Jinwook Seo, Wooshik Lee, Yeong-Gil Shin, and Bohyoung Kim. Automatic teeth axes calculation for well-aligned teeth using cost profile analysis

along teeth center arch. *IEEE Transactions on Biomedical Engineering*, 59(4):1145–1154, 2012. 3

[20] Automatic teeth axes calculation for well-aligned teeth using cost profile analysis along teeth center arch. *IEEE transactions on bio-medical engineering*, 59(4):1145, 2012. 3

[21] Qichang Hu, Huibing Wang, Teng Li, and Chunhua Shen. Deep cnns with spatially weighted pooling for fine-grained car recognition. *IEEE Transactions on Intelligent Transportation Systems*, 18(11):3147–3156, 2017. 5

[22] Yifei Shi, Junwen Huang, Hongjia Zhang, Xin Xu, Szymon Rusinkiewicz, and Kai Xu. Symmetrynet: learning to predict reflectional and rotational symmetries of 3d shapes from single-view rgb-d images. *ACM Transactions on Graphics (TOG)*, 39(6):1–14, 2020. 5

[23] Yinpeng Chen, Xiyang Dai, Mengchen Liu, Dongdong Chen, Lu Yuan, and Zicheng Liu. Dynamic convolution: Attention over convolution kernels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11030–11039, 2020. 5

[24] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 6

[25] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32:8026–8037, 2019. 6

[26] Ning Dai, Xiaoling Yu, Qilei Fan, Fulai Yuan, Lele Liu, and Yuchun Sun. Complete denture tooth arrangement technology driven by a reconfigurable rule. *PloS one*, 13(6):e0198252, 2018. 10

[27] Jin-gang Jiang, Ting Liang, Wei-ping Hu, et al. Kinematics modeling and experimentation of the multi-manipulator tooth-arrangement robot for full denture manufacturing. *Journal of medical systems*, 35(6):1421–1429, 2011. 10