Anatomically Constrained Deep Learning for Automating Dental CBCT Segmentation and Lesion Detection

Zhiyang Zheng, Hao Yan, Frank C. Setzer, Katherine J. Shi, Mel Mupparapu, and Jing Li, *Member, IEEE*

***Abstract*— Compared with the rapidly growing artificial intelligence (AI) research in other branches of healthcare, the pace of developing AI capacities in dental care is relatively slow. Dental care automation, especially the automated capability for dental cone beam computed tomography (CBCT) segmentation and lesion detection, is highly needed. CBCT is an important imaging modality that is experiencing ever-growing utilization in various dental specialties. However, little research has been done for segmenting different structures, restorative materials, and lesions using deep learning. This is due to multifold challenges such as content-rich oral cavity and significant within-label variation on each CBCT image as well as the inherent difficulty of obtaining many high-quality labeled images for training. On the other hand, oral-anatomical knowledge exists in dentistry, which shall be leveraged and integrated into the deep learning design. In this article, we propose a novel anatomically constrained Dense U-Net for integrating oral-anatomical knowledge with data-driven Dense U-Net. The proposed algorithm is formulated as a regularized or constrained optimization and solved using mean-field variational approximation to achieve computational efficiency. Mathematical encoding for transforming descriptive knowledge into a quantitative form is also proposed. Our experiment demonstrates that the proposed algorithm outperforms the standard Dense U-Net in both lesion detection accuracy and dice coefficient (DICE) indices in multilabel segmentation. Benefited from the integration with anatomical domain knowledge, our algorithm performs well with data from a small number of patients included in the training.**

***Note to Practitioners*—This article proposes a novel deep learning algorithm to enable the automated capability for cone beam computed tomography (CBCT) segmentation and lesion detection. Despite the growing adoption of CBCT in various dental specialties, such capability is currently lacking. The proposed work will provide tools to help reduce subjectivity and human errors, as well as streamline and expedite the clinical workflow. This will greatly facilitate dental care automation. Furthermore, due to the capacity of integrating oral-anatomical knowledge into the deep learning design, the proposed algorithm does not require many high-quality labeled images to train. The algorithm can provide good accuracy under limited training samples. This ability is highly desirable for practitioners by saving labor-intensive, costly labeling efforts, and enjoying the benefits provided by AI.**

***Index Terms*— Biomedical image segmentation, healthcare automation, machine learning, neural networks.**

I. Introduction

COMPARED with the rapidly growing artificial intelligence (AI) research in other branches of healthcare, the pace of developing AI capacities in dental care is relatively slow. Radiographic imaging is commonplace in dental care to assist clinicians in evaluation, diagnosis, and treatment planning. An important imaging modality called cone beam computed tomography (CBCT) is experiencing ever-growing utilization due to increased spatial resolution and 3-D imaging capability. CBCT has been used in a variety of dental fields such as endodontics, orthodontics, implant, oral surgery, and oral medicine [1].

In the various dental fields using CBCT, it is an important task for clinicians to accurately segment different structures, tissues, restorative materials, and lesions on each CBCT image. However, this capacity is currently lacking. Clinician-based interpretation of the CBCT images lacks precision, consistency, and objectivity, and thus suffering from low interobserver/intraobserver agreement [2]. Existing semiautomated computer-aided diagnosis (CAD) algorithms offer limited clinical utility as they are heavily dependent on clinicians for seed placement and manual adjustment to facilitate the image segmentation [3]–[9]. A significant amount of training is needed for clinicians to have the needed skill. Even with extensive training, human errors are common and inevitable. Also, because CBCT produces 3-D images, the scale of the data to be processed is overwhelming, which poses obstacles to the clinical workflow.

AI or deep learning holds great promise to provide a fully automated capability for CBCT analysis, which can help reduce subjectivity and errors. This capability can also help streamline and expedite the clinical workflow. However, the published work so far has focused on using deep learning to improve CBCT image quality [10], [11], facilitate reconstruction [12], and segment teeth [13]. Little research has been done for segmenting different structures (e.g., bone, teeth), restorative materials, and lesions. The reason for this gap

Manuscript received July 9, 2020; revised August 16, 2020; accepted September 18, 2020. Date of publication October 9, 2020; date of current version April 7, 2021. This article was recommended for publication by Lead Guest Editor A. Si and Editor M. Zhang upon evaluation of the reviewers' comments. This work was supported in part by the NSF DMS under Award 1830363 and Award 1903135. *(Corresponding author: Jing Li.)*

Zhiyang Zheng and Jing Li are with H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: zzheng93@gatech.edu; jing.li@isye.gatech.edu).

Hao Yan is with the School of Computing, Informatics, and Decision Systems Engineering, Arizona State University, Tempe, AZ 85281 USA (e-mail: haoyan@asu.edu).

Frank C. Setzer and Mel Mupparapu are with the School of Dental Medicine, University of Pennsylvania, Philadelphia, PA 19104 USA (e-mail: fsetzer@upenn.edu; mmd@upenn.edu).

Katherine J. Shi was with the University of Pennsylvania, Philadelphia, PA 19104 USA. She is now with the School of Dental Medicine, Tufts University, Boston, MA 02155 USA (e-mail: katherinejshi@gmail.com).

Color versions of one or more of the figures in this article are available online at https://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TASE.2020.3025871

is that CBCT segmentation is very challenging in multiple aspects: First, the oral cavity presented on each CBCT image is content-rich, containing different structures, tissues, restorative materials, and lesions—called labels hereafter in this article. Also, there is significant within-label variation. For example, materials can differ significantly in shape, size, intensity, and texture. Lesions also vary in size, shape, and so on. Furthermore, to train a deep learning algorithm to recognize the significant between- and within-label variations, there is a need for a large number of accurately labeled images by clinicians.

In general, deep learning algorithms need a large amount of labeled data to achieve high accuracy. However, in many healthcare applications, labeled data are limited due to the availability and cost of clinical experts. Also, high-quality labeled data may be inherently difficult to obtain due to the complexity of the problem domain. These are also the challenges faced by CBCT segmentation using deep learning. To address label data shortage in healthcare applications, existing research in deep learning has exploited different mechanisms such as transfer learning [14] and data augmentation [15]. More recently, research has also been done to develop semisupervised learning algorithms to incorporate unlabeled data in training, e.g., developing generative adversarial networks with semisupervision [16]. An alternative to these data-driven mechanisms is to integrate domain knowledge into the model training. Domain knowledge can be considered an auxiliary, special source of "data" to help boost the model performance. In addition, integration with domain knowledge prevents the generation of uninterpretable or counter-intuitive results.

In this article, we propose to integrate oral-anatomical knowledge with deep learning for CBCT segmentation and lesion detection. A commonly used deep learning algorithm for image segmentation is the U-Net. There have been some improvements of the traditional U-Net in recent years. We adopt the FC-Densenet [17] with some modifications as the baseline due to its parameter efficiency. We call this baseline network Dense U-Net. We further infuse anatomical knowledge as regularizations or constraints into the Dense U-Net design. We call the new algorithm "anatomically-constrained Dense U-Net." From oral anatomy, we have the knowledge regarding the relative locations of different structures, restorative materials, and specific types of lesions. Some examples of such knowledge include: a periapical lesion must be near the root of a tooth; restorative materials cannot be connected with the bone; since lesions and materials must attach to some structures or tissues, they cannot be surrounded by the background of the image. Incorporating anatomical knowledge has the effect of limiting the search space for the deep learning algorithm to find the optimal parameters. This has the potential to produce more accurate, interpretable results based on limited training data.

The contributions of this article are summarized as follows.

1) Contribution to the Methodology: Integration of domain knowledge and deep learning is a popular research area. It has been studied in natural language processing, solving of partial differential equations (PDEs), neural symbolic systems, molecular biology, and so on. Section II includes a detailed review of the existing work in this field. However, no work has been done for integrating oral-anatomical knowledge with dental image segmentation to our best knowledge. The proposed anatomically constrained Dense U-Net provides the first-of-its-kind framework of the integration. The technical novelties of the proposed method include the following.

1) We propose a regularized optimization framework that aims to minimize the loss function on training data, while at the same time striving for the consistency with oral-anatomical knowledge in expectation.
2) We propose a mathematical encoding of the knowledge, which is descriptive in its original form, into quantitative pixel-wise consistency functions. This quantitative form makes it possible for the knowledge to be integrated with deep learning.
3) To resolve the computational challenge of incorporating hundreds of thousands of pixel-wise constraints—equal to the size of an CBCT image, we propose to use variational inference and mean-field approximation to produce a tractable solution for the optimization.

2) Contribution to Dental Care Automation: Our experiment demonstrates that the proposed anatomically constrained Dense U-Net outperforms the data-driven Dense U-Net in both lesion detection accuracy and voxel-matching dice coefficient (DICE) accuracy for each label. Our algorithm can achieve good performance with data from a small number of patients included in the training. This demonstrates the value of the integration with anatomical domain knowledge. This is the first work that uses deep learning for CBCT multilabel segmentation and periapical lesion detection. This work demonstrates the potential of using AI to automate dental care. Given that CBCT is popularly used in various dental specialties, developing AI capabilities for CBCT will profoundly impact dental practices and patient care.

The remainder of this article is organized as follows. Section II reviews related work. Section III presents the proposed anatomically constrained Dense U-Net. Section IV presents the experiment and results. Section V is the conclusion.

II. Related Works

From the methodological point of view, this article is related to the field in deep learning that investigates how to integrate domain knowledge with data-driven algorithms (Section II-A) and the field of U-Net and its improvements (Section II-B). From the application point of view, this article is related to deep learning applications for CBCT (Section II-C). In what follows, we will review the existing work related to both the methodology and applications. Finally, we will point out the gap in the existing research and the need for the proposed work (Section II-D).

A. Integration of Domain Knowledge With Deep Learning

Integration of domain knowledge with deep learning algorithms has been mainly investigated for three types of knowledge: relational knowledge, physical knowledge, and logical knowledge.

Relational knowledge includes simple relations such as father and son, as well as more complicated, structured forms encoded by knowledge graphs or statistical relational models. The integration of relational knowledge with deep learning has been mainly investigated in language-related tasks, including text translation, text comprehension, and knowledge graphs. Most of the existing work used attention mechanisms [18]–[20], multihop architectures [21], [22], or their combinations [23] in recurrent neural networks (RNNs). Other research combined prior knowledge explicitly into RNN. For example, GRAFT-Net used the knowledge base as a prior source and a graph representation learning convolutional neural network (CNN) to infuse the graph-like knowledge base information into an RNN model [24].

Integrating physical knowledge with deep learning has mostly been studied within the context of PDEs. Deep Galerkin method (DGM), a deep learning model inspired by the Galerkin method of numerically solving PDE, aimed to use deep neural networks to reach approximate numerical solutions of PDE [25]. Other research, such as PDE-Net, aimed to use deep learning algorithms to uncover hidden PDE formulations [26].

Logical knowledge is a typical form of human knowledge, which can be coded by first-order logical rules or probabilistic graphical models [27]. Integrating logical knowledge with deep learning has been done in several ways. Neural symbolic systems represented a type of integration [28], in which neural networks were designed on given rules to perform reasoning, such as KBANN [29] and CILP++ [30]. Grammar variational autoencoder used grammar rules to encode discrete data, e.g., moleculars, to parse trees [31]. Other than these problem-specific designs, some researchers tried to make a more general model for the integration by encoding the knowledge as features or other formats easily transferred to a neural network. Collobert *et al.* [32] proposed an approach to extract extra features from knowledge in texts. Karaletsos *et al.* [33] proposed to express similarity rules as a triplet format and transfer them to a Bayesian latent factor model. Some researchers used posterior regularization to infuse domain knowledge to deep learning models [34], [35].

B. U-Net and Its Improvements

In medical image segmentation, one of the most commonly used algorithms is U-Net [36]. This network is based on a symmetric encoder–decoder framework and widely used because of its capability to train with limited data. The most important advantage of U-Net is the introduction of skip connection to CNNs, which helped the network to retrieve spatial information lost in downsampling procedures. Also, U-Net adopts 2×2 transposed convolution operation to perform upsampling in the decoder part, which makes the training of upsampling possible.

The conventional U-Net has been improved by fusing with other network structures in recent years. For example, FC-Densenet [17] integrated the idea of dense blocks in Dense-Net [37] into U-Net design. It changed the convolution layers at each level to a dense block, which took advantage of fewer parameters. QuickNAT [38] used the structure of FC-Densenet with different settings and substituted the transposed convolutional layers with un-pooling layers [39]. MultiRes U-Net [40] used residual connection [41] at every level of U-Net and used residual paths to replace skip connection, which made a deeper network with better performance. With the same intuition as MultiRes U-Net, U-Net++ [42] used dense connection layers to replace skip connection, which retained more spatial information.

C. Deep Learning Applications for CBCT

Deep learning has been used to map CBCT to high-quality CT images [10], [11]. Research has also been done to facilitate the image reconstruction process [12]. For image segmentation, existing work has focused on segmenting and classifying teeth. For example, Miki *et al.* [43] proposed to use Alex-Net to classify teeth to different types. Pavaloiu *et al.* [44] used basic neural networks for edge detection. More recently, Zakirov *et al.* [45] used V-Net to segment teeth on 3-D images and used a fully connected neural network (FCNN) to further classify the teeth.

D. Gaps in the Existing Research

On the application side, little work has been done for CBCT segmentation and lesion detection using deep learning. This is due to the multifold challenges of this task, such as content-rich oral cavity and significant within-label variation on each CBCT image as well as the inherent difficulty of obtaining many high-quality labeled images for training. Integration of domain knowledge about oral anatomy with deep learning holds great promise to resolve these challenging issues. However, there is a lack of methodological development in deep learning to achieve such integration. This article aims to bridge this gap.

III. Proposed Anatomically Constrained Dense U-Net

In this section, we will first present our modified design of the existing FC-Densenet to fit our problem setting, called Dense U-Net, in Section III-A. Then, we will present the proposed optimization framework for integrating oral-anatomical knowledge and Dense U-Net in Section III-B. In Section III-C, we will present how to mathematically encode anatomical knowledge. Finally, we will propose an efficient algorithm to solve the optimization framework and obtain probabilistic label maps for CBCT images in Section III-D.

A. Dense U-Net Design

From several improved U-Net structures, we select the structure of FC-Densenet [17] due to its parameter efficiency. However, the original FC-Densenet has over 100 layers, which still has too many parameters for our small data set and runs the risk of over-fitting. Thus, we make some modifications to this structure and call it Dense U-Net. The Dense U-Net design has five components, such as the initial convolutional layer, dense blocks, transition down blocks, transition up blocks, and the output convolutional layer.

A dense block has several repeated layers. Each layer has a batch normalization, rectified linear unit (ReLu) activation, a 3×3 convolutional layer, and a dropout layer. The layers in a dense block have a dense connection, which means that the input of one layer is a concatenation of outputs of all previous layers. A dense block has a growth rate k, which represents the number of output channels of every layer in the dense block.

A transition down block includes batch normalization, ReLu activation, a 1×1 convolutional layer, a dropout layer, and a 2×2 max-pooling layer. It takes the output of a dense block and performs downsampling. The first part of a transition-down block before max-pooling can be used to perform feature compression, which can significantly reduce the number of parameters. The second part of a transition-down block is a max-pooling layer for downsampling.

A transition-up block is a 2×2 transposed convolution with stride 2, which is an upsampling process that can be trained. The upsampled map is concatenated to the output of the previous dense block at the same level, which retains the spatial information.

The workflow of Dense U-Net in use is the following: an image that is put into Dense U-Net will be processed by an initial 3×3 convolutional layer at first. Then, the output will be processed by a downsampling path with several dense blocks and transition down blocks. The upsampling path will then process the encoded feature map with skip connection from the output of dense blocks in the downsampling path at each level. Finally, a 1×1 convolutional layer will process the output of the upsampling path to generate the segmentation map.

B. Optimization Framework for Integrating Oral Anatomy and Dense U-Net Using Variational Inference

Let $\mathbf{x}$ denote a CBCT image of P pixels, i.e., $\mathbf{x} = \{x_j; j = 1, \ldots, P\}$, where x_j is the image intensity at the jth pixel. $P = r \times c$, where r and c are the numbers of rows and columns of a CBCT image, respectively. Let $\mathbf{y}$ contain the pixel labels with the same size of $\mathbf{x}$, i.e., $\mathbf{y} = \{y_j; j = 1, \ldots, P\}$, where y_j is the label of the jth pixel. Assume K possible labels for each pixel. The goal of an image segmentation task is to train an algorithm to map $\mathbf{x}$ to $\mathbf{y}$. Dense U-Net builds a probabilistic mapping between $\mathbf{x}$ and $\mathbf{y}$, $p_\theta(\mathbf{y}|\mathbf{x})$. θ contains parameters of the Dense U-Net.

To train the parameters of a Dense U-Net, a set of N labeled images will be used. Denote this training set by $\boldsymbol{D}_l$. The parameters are learned to minimize a loss function averaged over the training samples, i.e., $(1/N)\sum_{(\mathbf{x},\mathbf{y})\in\boldsymbol{D}_l} \mathcal{L}(\mathbf{y}, p_\theta(\mathbf{y}|\mathbf{x}))$. Commonly used loss functions include the multiclass cross-entropy loss, the focal loss [48], and the multiclass cross-entropy loss plus DICE loss [39], [41].

However, the standard Dense U-Net is purely data-driven, i.e., it does not consider oral-anatomical knowledge. To incorporate the knowledge, we propose a regularized loss function to balance between minimizing the data-driven loss (e.g., cross-entropy, focal, and cross-entropy plus DICE losses as mentioned above) and maximizing the consistency with knowledge in expectation, i.e.,

$$\min_\theta \left\{ \begin{array}{l} \frac{1}{N}\sum_{(\mathbf{x},\mathbf{y})\in\boldsymbol{D}_l} \mathcal{L}(\mathbf{y}, p_\theta(\mathbf{y}|\mathbf{x})) - \\ \alpha \frac{1}{N}\sum_{\mathbf{x}\in\boldsymbol{D}_l} E_{p_\theta(\mathbf{y}|\mathbf{x})}(f(\mathbf{y})) \end{array} \right\}. \tag{1}$$

The second term in (1) corresponds to the knowledge. $f(\mathbf{y})$ is a function of the pixel labels in $\mathbf{y}$ for a CBCT image, which reflects the consistency of the pixel labels with respect to the knowledge. The definition of this consistency function $f(\mathbf{y})$ depends on the type of knowledge, which will be discussed with more detail in Section III-C. Here, we focus on the general notation in order to better describe the overall framework. Since $f(\mathbf{y})$ is a random variable, we take the expectation of it on the conditional distribution of $p_\theta(\mathbf{y}|\boldsymbol{x})$. By doing this, we encourage the consistency with knowledge on an "average" sense. Also, we encourage this consistency over a set of images. In (1), we assume that this set of images includes the N labeled images. However, in the general case, this set can include not only labeled images but also unlabeled images, because we do not need to know the labels in $\mathbf{y}$ but only the expectation of $f(\mathbf{y})$. α is a tuning parameter.

It is difficult to evaluate the expectation of $f(\mathbf{y})$, because $p_\theta(\mathbf{y}|\boldsymbol{x})$ is a joint distribution of the labels for all pixels on a CBCT image, which has a huge dimension and is extremely computationally intensive. To resolve this issue, we propose to borrow the idea from variational inference. Variational inference was originally developed as an approach to approximate a difficult-to-compute posterior distribution in Bayesian statistics, which finds an approximate distribution in a variational family to minimize the Kullback–Leibler (KL) divergence to the exact posterior [46]. The variational family is typically chosen for computational benefits, such as the commonly used mean-field variational family [47]. Our problem is not Bayesian, but we borrow the idea of variational inference and seek an approximate distribution $q(\mathbf{y}|\mathbf{x})$ for $p_\theta(\mathbf{y}|\mathbf{x})$. In our case, $q(\mathbf{y}|\mathbf{x})$ is found not only to minimize the KL divergence with respect to $p_\theta(\mathbf{y}|\mathbf{x})$, like in the original variational inference, but also to maximize the consistency with knowledge in expectation, i.e.,

$$\min_q \mathrm{KL}(q(\mathbf{y}|\mathbf{x})||p_\theta(\mathbf{y}|\mathbf{x})) - \lambda E_{q(\mathbf{y}|\mathbf{x})}(f(\mathbf{y})). \tag{2}$$

λ is a tuning parameter.

Furthermore, combining (1) and (2), we can get a new optimization problem, i.e.,

$$\min_{\theta,q} \left\{ \begin{array}{c} \frac{1}{N}\sum_{(\mathbf{x},\mathbf{y})\in\boldsymbol{D}_l} \mathcal{L}(\mathbf{y}, p_\theta(\mathbf{y}|\mathbf{x})) - \\ \alpha \frac{1}{N}\sum_{\mathbf{x}\in\boldsymbol{D}_l} \left(\lambda E_{q(\mathbf{y}|\mathbf{x})}(f(\mathbf{Y})) - \mathrm{KL}(q(\mathbf{y}|\mathbf{x})||p_\theta(\mathbf{y}|\mathbf{x}))\right) \end{array} \right\}. \tag{3}$$

The benefit of introducing the variational distribution $q(\mathbf{y}|\mathbf{x})$ can be better revealed as follows: given θ, (3) becomes (2), which can be solved in an analytical form, i.e.,

$$q^*(\mathbf{y}|\mathbf{x}) = \frac{1}{C} p_\theta(\mathbf{y}|\mathbf{x}) \exp\{\lambda f(\mathbf{y})\} \tag{4}$$

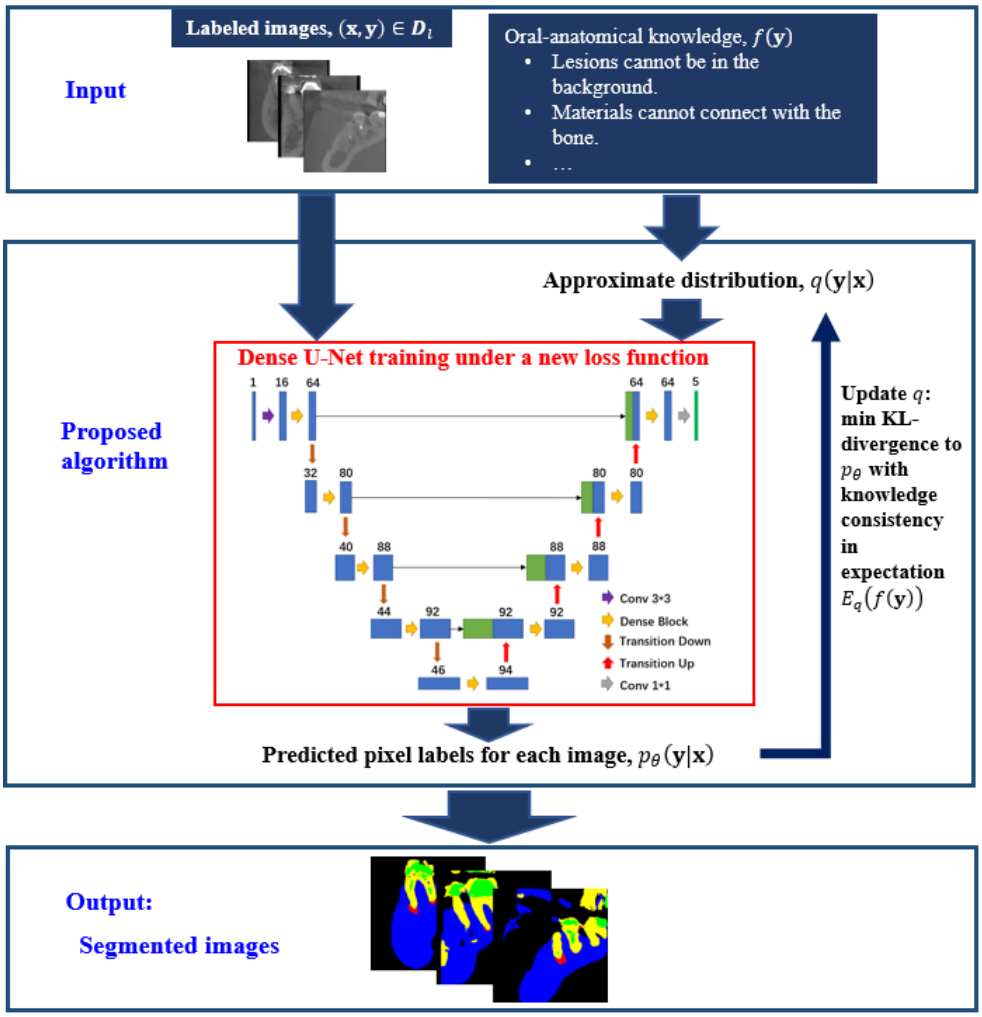


Fig. 1. Schematic overview of the proposed anatomically constrained Dense U-Net. The workflow is as follows: the network takes a set of labeled images and anatomical knowledge as input. To initialize, Dense U-Net is first used to produce initial values for the parameters, θ, based on images, and generate predicted pixel labels, $p_\theta(y|x)$. Then, two optimizations are iteratively solved between updating θ and $q(y|x)$: 1) given θ, $q(y|x)$ is updated by minimizing KL-divergence with $p_\theta(y|x)$ and maximizing consistency with anatomical knowledge and 2) given $q(y|x)$, θ is updated by minimizing the data-driven loss and KL-divergence with $q(y|x)$. Details of these optimizations are discussed in Section III-D. The output from the network contains segmented images.

where C is a normalizing constant. Furthermore, given q, (3) becomes

$$\min_{\theta} \times \left\{ \frac{1}{N} \sum_{(\mathbf{x},\mathbf{y}) \in \boldsymbol{D}_l} \mathcal{L}(\mathbf{y}, p_\theta(\mathbf{y}|\mathbf{x})) + \alpha \frac{1}{N} \sum_{\mathbf{x} \in \boldsymbol{D}_l} \mathrm{KL}(q(\mathbf{y}|\mathbf{x}) || p_\theta(\mathbf{y}|\mathbf{x})) \right\}$$

which can be treated as a new loss function optimized by the Dense U-Net. Note that this process of iteratively solving the optimization in (3) avoids the need for evaluating the expectation of $f(\mathbf{y})$—a challenge we have mentioned previously to be computationally intractable. Please see Fig. 1 for a schematic overview of the proposed optimization framework.

Next, we would like to reveal some insight about the impact of incorporating knowledge. From (4), we can see that the data-driven distribution $p_\theta(\mathbf{y}|\mathbf{x})$ is modified by incorporating the consistency function with knowledge, $f(\mathbf{y})$, to become $q^*(\mathbf{y}|\mathbf{x})$. When $\mathbf{y}$ takes on values/labels that yield a larger $f(\mathbf{y})$, i.e., a higher consistency with knowledge, the corresponding $q^*(\mathbf{y}|\mathbf{x})$ will be increased from $p_\theta(\mathbf{y}|\mathbf{x})$ by a larger factor to make these values of $\mathbf{y}$ more likely. In other words, the probabilities of different values of $\mathbf{y}$ are rearranged in $q^*(\mathbf{y}|\mathbf{x})$ to reflect their respective consistency with knowledge.

Finally, we would like to point out that despite the close-form solution in (4), it is still difficult to compute $q^*(\mathbf{y}|\mathbf{x})$ due to the high dimensionality of $\mathbf{y}$. Take the images in our experiment as an example. There are 256×256 pixels on a CBCT image and five labels for each pixel, {"lesion," "bone," "teeth," "materials," "background"}. This results in a total of $5^{256\times256}$ combinations of pixel labels, which make it impossible to compute the normalizing constant. To resolve this issue, we propose to use the mean-field variational family for $q(\mathbf{y}|\mathbf{x})$ to produce a tractable and efficient solution, which will be discussed with more detail in Section III-C.

C. Mathematical Encoding of Anatomical Knowledge

In Section III-B, we referred to $f(\mathbf{y})$ as a consistency metric with respect to the anatomical knowledge. In general,

the definition of $f(\mathbf{y})$ is flexible and depends on the type of knowledge. In this article, we focus on the knowledge about the relative position of different labels according to human oral anatomy. Specifically, we focus on the knowledge stating that the segment of an image belonging to label k cannot be in/connect with another segment belonging to label k'. For example, since lesions (i.e., label k) must attach to some structure or tissue, it cannot be in the background (i.e., label k'). Restorative materials (i.e., label k) cannot connect with the bone (i.e., label k'). In our experiment, we find these two pieces of knowledge regarding the lesions and materials are particularly helpful, because these two labels are most difficult to segment among all the labels in a CBCT image, due to significant within-label variation in shape, size, intensity, and texture.

To transform the descriptive knowledge into a quantitative form, we propose to examine the labels of the pixels that are neighbors of each other. For example, if a pixel is labeled as "lesion," we should penalize the likelihood for the neighboring pixels to be labeled as "background." This has the effect of biasing the model training to be consistent with the knowledge that a lesion cannot be in the background. To realize this idea in a mathematical form, we first decompose $f(\mathbf{y})$ into pixel-wise consistency metrics to facilitate the checking the label of each pixel against its neighbors, i.e.,

$$f(\mathbf{y}) = \sum_{j=1}^{P} f_j\left(y_j, \mathbf{y}_{\mathrm{NE}(j)}\right) \tag{5}$$

where $f_j(\cdot)$ corresponds to each pixel in the CBCT image and only involves the pixel y_j and its neighboring pixels $\mathbf{y}_{\mathrm{NE}(j)} = \{y_i; i \in \mathrm{NE}(j)\}$. Here, the neighborhood of a pixel, $\mathrm{NE}(j)$, is user-defined, which typically includes the four immediate neighbors but can include more pixels. Furthermore, to encode the knowledge that the segment belonging to label k cannot be in/connect with another segment belonging to label k', we further define $f_j\left(y_j, \mathbf{y}_{\mathrm{NE}(j)}\right)$ as

$$f_j\left(y_j, \boldsymbol{y}_{\mathrm{NE}(j)}\right) = \begin{cases} 1, & y_j \neq k \\ \prod\limits_{i \in NE(j)} I\left(y_i \neq k'\right), & y_j = k. \end{cases} \tag{6}$$

The meaning of (6) can be understood as follows: if y_j is label k (e.g., "lesion"), the consistency $f_j\left(y_j, \mathbf{y}_{\mathrm{NE}(j)}\right)$ is the maximum with a value of one only when none of the pixels in the neighborhood is label k' (e.g., "background"). If y_j is not label k, $f_j\left(y_j, \mathbf{y}_{\mathrm{NE}(j)}\right)$ becomes irrelevant to this particular knowledge regarding label k and the consistency is automatically achieved.

Suppose there are Ω pieces of knowledge of this kind to be integrated with deep learning. Let $f_j^{(k)}\left(y_j, \boldsymbol{y}_{\mathrm{NE}(j)}\right)$, $k \in \Omega$, be the pixel-wise consistency metric with respect to each piece of knowledge in Ω. We can sum these consistency metrics together to become an overall consistency with respect to the collective knowledge set, i.e.,

$$\sum_{k \in \Omega} \lambda_k f_j^{(k)}\left(y_j, \boldsymbol{y}_{\mathrm{NE}(j)}\right)$$

where λ_k is the weight for each piece of knowledge.

D. Efficient Algorithm for Solving the Anatomically Constrained Dense U-Net Optimization Using the Mean-Field Approximation

Recall that at the end of Section III-A, we pointed out the challenge of computing $q^*(\mathbf{y}|\mathbf{x})$ in (4) due to the high dimensionality of $\mathbf{y}$. To resolve this issue, we propose to choose a specific form of q from the mean-field variational family [37] to alleviate the computational complexity. The mean-field family assumes that q can be represented by a product of pixel-wise distributions q_j's, i.e.,

$$q(\mathbf{y}|\mathbf{x}) = \prod_{j=1}^{P} q_j\left(y_j|\mathbf{x}\right). \tag{7}$$

Using the $q(\mathbf{y}|\mathbf{x})$ in (8), as shown at the bottom of the next page, and the $f(\mathbf{y})$ specified in Section III-B, the optimization in (4) becomes The benefit of this optimization formulation is that given $\theta, q_1, \ldots, q_P$ can be solved iteratively. This makes the computation tractable and efficient. Furthermore, given $q_1, \ldots, q_P, \theta$ can be solved by the standard stochastic gradient algorithm. These two steps can be iterated to get the final solution for (8). Next, we discuss the specifics of each step.

Given θ, the optimization in (8) becomes

$$\min_{q_1,\ldots,q_P} \frac{1}{N} \sum_{\mathbf{x} \in \boldsymbol{D}_l} \begin{pmatrix} \mathrm{KL}\left(\prod_{j=1}^{P} q_j\left(y_j|\mathbf{x}\right) || p_\theta(\mathbf{y}|\mathbf{x})\right) - \\ E_{q_1,\ldots,q_P}\left(\sum_{j=1}^{P} \sum_{k \in \Omega} \lambda_k f_j^{(k)}\left(y_j, \boldsymbol{y}_{\mathrm{NE}(j)}\right)\right) \end{pmatrix} \tag{9}$$

which can be solved iteratively over $q_1, \ldots, q_P$ according to Proposition 1.

Proposition 1: Given $q_1^{(t-1)}, \ldots, q_{j-1}^{(t-1)}, q_{j+1}^{(t-1)}, \ldots, q_P^{(t-1)}$ obtained from the $(t-1)$th iteration, the optimization with respect to q_j at the tth iteration, $q_j^{(t)}$, can be solved by

$$q_j^{(t)} = \frac{1}{c_j^{(t)}} p_\theta\left(y_j|\mathbf{x}\right) \exp\left(\sum_{k \in \Omega} \lambda_k g_j^{(k)}\left(y_j\right)\right)$$

where

$$g_j^{(k)}\left(y_j\right) = \begin{cases} 1, & y_j \neq k \\ \prod\limits_{i \in \mathrm{NE}(j)} q_i^{(t-1)}\left(y_i \neq k'\right), & y_j = k \end{cases} \tag{10}$$

and $c_j^{(t)}$ is the normalizing constant.

Please see Appendix for the proof and calculation of $c_j^{(t)}$. Using Proposition 1, $q_j^{(t)}$ can be updated for one pixel at a time. This makes the normalizing constant easy to compute.

Furthermore, given $q_1, \ldots, q_P$, (9) becomes

$$\begin{aligned} &\min_\theta L(\theta) \\ &\triangleq \min_\theta \left\{ \begin{array}{c} \frac{1}{N} \sum_{(\mathbf{x},\mathbf{y}) \in \boldsymbol{D}_l} \mathcal{L}(\mathbf{y}, p_\theta(\mathbf{y}|\mathbf{x})) + \\ \alpha \frac{1}{N} \sum_{\mathbf{x} \in \boldsymbol{D}_l} \mathrm{KL}\left(\prod_{j=1}^{P} q_j\left(y_j|\mathbf{x}\right) || p_\theta(\mathbf{y}|\mathbf{x})\right) \end{array} \right\} \end{aligned} \tag{11}$$

where $L(\theta)$ can be treated as a new loss function optimized by the Dense U-Net and thus can be solved by the standard stochastic gradient algorithm. Specifically, we can randomly

sample a batch of samples from $\boldsymbol{D}_l$ to compute the gradient of $L(\theta)$ to update θ accordingly, i.e.,

$$\theta^{(t)} = \theta^{(t-1)} - \frac{\partial}{\partial\theta} L(\theta). \tag{12}$$

IV. Experiment

A. Data Description

Our data set consists of 20 patients with periapical lesions collected from the School of Dental Medicine, University of Pennsylvania, Philadelphia, PA, USA. Institutional review board (IRB) approval was obtained prior to the study. CBCTs of all the patients were acquired by a Morita Veraviewpocs 3DF40 field of view (FOV) 40 mm at voxel size 125-mm machine. The CBCT of each patient includes limited FOV containing roots with at least one lesion.

To prepare labeled data for training, ITK-SNAP was utilized to assist the clinicians to generate manual segmentation on the CBCT volumes following protocols established in previous studies [51]–[53]. From raw volumes, five standardized categories of image content were labeled: "lesion," "bone," "teeth," "materials," "background." ITK-SNAP uses a semiautomatic segmentation algorithm, which requires the manual seed selection to start with. The algorithm will then automatically evolve the initial seed to the area of interest. After the algorithm segmentation, the segmentation was reviewed and manually revised by three reviewers (dental experts). All reviewers were calibrated and trained in CBCT analysis and segmentation and included an oral and maxillofacial radiologist (M.M.), an endodontist (F.C.S.), and a senior graduate student completing a radiology honors program (K.J.S.). Disagreements were resolved by joint discussion. Because manual segmentation is extremely time-consuming (up to several hours per CBCT) and was, therefore, not carried out by each individual reviewer, it was a joint review process.

However, we can use lesion detection for an estimate of the consistency between the three reviewers. Lesion detection was carried out using the raw CBCT images before any joint segmentation process. The result is the following: the initial agreement between the reviewers was 27/29 (or 93.1% agreement) for roots with lesions and 100% for roots with no lesions. The 2/29 discrepancies were resolved by joint review and discussion.

Five slices in the sagittal view from each CBCT were included in this study, which resulted in a total of $20 \times 5 = 100$ images to include in the training. Each image contains 256×256 pixels.

B. Image Preprocessing

To standardize the contrast across all the images, the contrast curve of each image was linearly adjusted to be between -5000 and 10 000 range of intensity. To fully utilize the small-size data set and avoid over-fitting, data augmentation was applied. A random image generator was used, with the rotation of degree range 0.2, width and height shifting by 5%, intensity sheared by 5%, zooming by 5%, and horizontal, vertical flip. All these augmentation processes and parameters were randomly chosen inside the range in each batch during the training.

C. Model Setup and Training

The specific Dense U-Net setting used in this study is the following: at the initial convolutional layer, the network takes images as input and produces a feature map with 16 channels. Each dense block consists of four convolutional layers, and the growth rate k is set to 12. In transition down blocks, there is a 1^*1 convolution layer for feature compression, with the compression rate set to be 0.5. The dropout rate of all dropout layers is set to 0.2 as previous studies [17], which can help avoid over-fitting.

A fourfold cross-validation was performed on the 20 patients. Each time, the CBCT images of 15 patients were used in training, and the model was validated on the remaining five patients. This process was iterated over all the folds. The model was implemented in TensorFlow using the Keras module [50]. Batch training was used with two images in a batch. The Adam optimizer was chosen with a learning rate of 8×10^{-5}. The model was initialized with the Glorot uniform initializer as default in TensorFlow. The model was trained with 100 epochs, 40 batches in each epoch, and 2 images in each batch.

Two pieces of anatomical knowledge regarding lesions and materials were integrated with the data in training: {lesions cannot be in the background; materials cannot connect with the bone}. Although other types of knowledge can be included, we found that these two pieces of knowledge helped the most because materials and lesions are most difficult to segment.

There are several parameters to tune. Next, we discuss the detailed procedure and considerations. λ_1 and λ_2 are the weights corresponding to the two pieces of knowledge. To select them in a computationally efficient way, we performed a two-phase grid search. The first phase was a coarse search between [1, 5] on integer values. The result of this phase narrowed down the ranges of λ_1 and λ_2 to [2, 3] and [1, 2], respectively. The second phase was a fine-scaled search within the ranges identified in the first phase, which found $\lambda_1 = 2.8$, $\lambda_2 = 1.0$ yielded the best performance.

Additionally, considering that the size of each label varies significantly across different labels, we used different weights for different labels in the cross-entropy loss to address the sample imbalance. The weights were roughly chosen to be

$$\min_{\theta, q_1, \ldots, q_p} \left\{ \begin{array}{c} \frac{1}{N} \sum_{(\mathbf{x},\mathbf{y}) \in \boldsymbol{D}_l} \mathcal{L}(\mathbf{y}, p_\theta(\boldsymbol{y}|\mathbf{x})) - \\ \alpha \frac{1}{N} \sum_{\mathbf{x} \in \boldsymbol{D}_l} \left(\begin{array}{l} E_{q_1,\ldots,q_P}\left(\sum_{j=1}^{P} \sum_{k \in \Omega} \lambda_k f_j^{(k)}\left(y_j, \boldsymbol{y}_{\mathrm{NE}(j)}\right) \right) - \\ \mathrm{KL}\left(\prod_{j=1}^{P} q_j\left(y_j|\mathbf{x}\right) || p_\theta(\mathbf{y}|\mathbf{x}) \right) \end{array} \right) \end{array} \right\}. \tag{8}$$

inversely proportional to the size of each label, i.e., the label of a smaller size (e.g., lesion) was given a larger weight, while the label of a larger size (e.g., background) was given a smaller weight. These weights are used in weighted cross-entropy loss. For focal loss, its parameter γ is set to 1, as used in [48].

Furthermore, there is a tuning parameter α in the optimization in (11), which controls the tradeoff between the cross-entropy loss on labeled images and the KL divergence between the distributions q and p_θ. The following formula was used to make α a changing parameter over the training iterations:

$$\alpha^{(t)} = \begin{cases} 0, & t < 50 \\ \min\left(1 - 0.9^{t-50}, 0.5\right), & t \geq 50 \end{cases}$$

and (11) is adjusted to be

$$\min_{\theta} \left\{ \begin{array}{l} \left(1-\alpha^{(t)}\right)\frac{1}{N}\sum_{(\mathbf{x},\mathbf{y})\in \boldsymbol{D}_l} \mathcal{L}(\mathbf{y}, p_\theta(\mathbf{y}|\mathbf{x})) + \\ \alpha^{(t)}\frac{1}{N}\sum_{\mathbf{x}\in \boldsymbol{D}_l} KL\left(\prod_{j=1}^{P} q_j\left(y_j|\mathbf{x}\right)||p_\theta(\mathbf{y}|\mathbf{x})\right) \end{array} \right\}.$$

The reason is as follows: the estimated q distribution is inaccurate at the beginning of the training. To consider this, α at the beginning of the training (i.e., small t) will be 0 according to the formula, which accounts less for the KL divergence. As the training goes on and after a number of epochs, the estimation for q gets better, the weight of the KL divergence will rise with increasing epochs and finally reach 0.5 to keep a balance between the impacts of labeled data and the knowledge.

D. Evaluation Metrics

We compared the results of the proposed anatomically constrained Dense U-Net with the standard Dense U-Net under three loss functions: cross-entropy loss, focal loss, and multiclass logistic loss plus DICE loss. We adopted several evaluation metrics. The first metric is per-root lesion detection accuracy. As a periapical lesion is close to the root of a tooth, we had our clinical collaborators identify the root on each CBCT image. Then, for each root, if there is a lesion in both the manual segmentation and the segmentation by a deep learning algorithm, it is counted as a "match." A "match" is also counted if both the manual and the algorithm's segmentations agree that there is no lesion for a root. A "miss" is counted if there is disagreement. In this way, we can compute a confusion matrix and calculate precision and recall that reflect lesion detection accuracy for each algorithm.

Furthermore, to evaluate the accuracy of each algorithm in the segmentation of each label, we computed the DICE index [49]. DICE has been widely used to evaluate pixel-level label matching between manual segmentation (considered as ground truth) and the segmentation by an algorithm. To compute DICE for a label k (e.g., lesion), let Y_k and $\hat{Y}_k$ be the sets of pixels on an image belong to label k by manual segmentation and by an algorithm, respectively. Then, DICE_k is computed by

$$\text{DICE}_k = \frac{2\left|Y_k \bigcap \hat{Y}_k\right|}{|Y_k| + |\hat{Y}_k|}.$$

TABLE I
PRECISION AND RECALL OF PER-ROOT LESION DETECTION

	Precision	Recall
Dense U-Net (cross-entropy loss)	0.65	0.7
Anatomically-constrained Dense U-Net (cross-entropy loss & anatomical constraints)	**0.9**	**0.8**
P-value of difference	<0.001*	0.042*
Dense U-Net (focal loss)	0.68	0.8
Anatomically-constrained Dense U-Net (focal loss & anatomical constraints)	**0.83**	**0.82**
P-value of difference	0.008*	0.469
Dense U-Net (multi-class logistic loss plus DICE loss)	0.81	0.78
Anatomically-constrained Dense U-Net (multi-class logistic loss plus DICE loss & anatomical constraints)	**0.9**	**0.84**
P-value of difference	0.039*	0.128

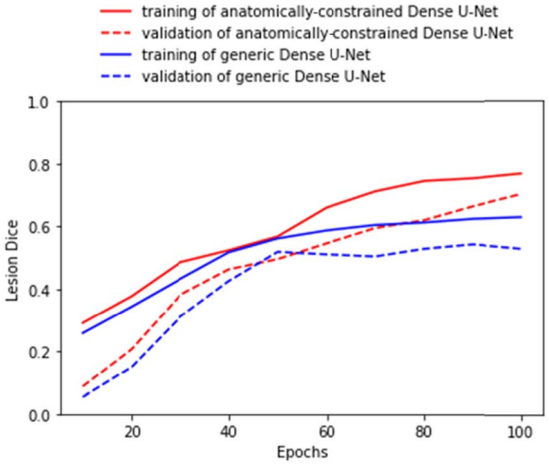


Fig. 2. Learning curves of lesion DICE.

E. Results and Discussion

The results of lesion detection accuracy and DICE are shown in Tables I and II. Fig. 2 shows the learning curves. Fig. 3 shows several examples of the segmented images.

From Tables I and II, we can observe that the proposed anatomically constrained Dense U-Net generally improves lesion detection accuracy and DICE indices. Specifically, from Table I, we can observe that under all three loss functions (i.e., cross-entropy loss, focal loss, and multiclass logistic plus Dice loss), the proposed anatomically constrained Dense U-Net has significantly better precision than the data-driven Dense U-Net. The proposed algorithm also has significantly better recall under the cross-entropy loss. From Table II, we can conclude that the proposed anatomically constrained Dense U-Net has significantly better DICE for lesion, the most important label, under all three loss functions. The proposed algorithm also shows better DICE for other labels.

TABLE II
DICE FOR PIXEL-LEVEL MATCHING ACCURACY

	Background	Lesion	Material	Bone	Teeth
Dense U-Net (cross-entropy loss)	0.957	0.606	**0.817**	0.856	**0.787**
Anatomically-constrained Dense U-Net (cross-entropy loss & anatomical constraints)	**0.958**	**0.672**	0.814	**0.87**	0.782
Std of difference	0.013	0.234	0.189	0.061	0.05
P-value of Wilcoxon Test	0.362	<0.001*	0.962	<0.001*	0.582
	Background	**Lesion**	**Material**	**Bone**	**Teeth**
Dense U-Net (focal loss)	0.947	0.589	0.82	0.844	0.766
Anatomically-constrained Dense U-Net (focal loss & anatomical constraints)	**0.959**	**0.709**	**0.822**	**0.877**	**0.801**
Std of difference	0.02	0.28	0.155	0.074	0.067
P-value of Wilcoxon Test	<0.001*	<0.001*	0.067	<0.001*	<0.001*
	Background	**Lesion**	**Material**	**Bone**	**Teeth**
Dense U-Net (multi-class logistic loss plus DICE loss)	0.96	0.691	0.805	0.881	0.798
Anatomically-constrained Dense U-Net (multi-class logistic loss plus DICE loss & anatomical constraints)	**0.961**	**0.741**	**0.818**	**0.887**	**0.81**
Std of difference	0.014	0.225	0.111	0.059	0.06
P-value of Wilcoxon Test	0.153	0.022*	0.351	<0.001*	0.247

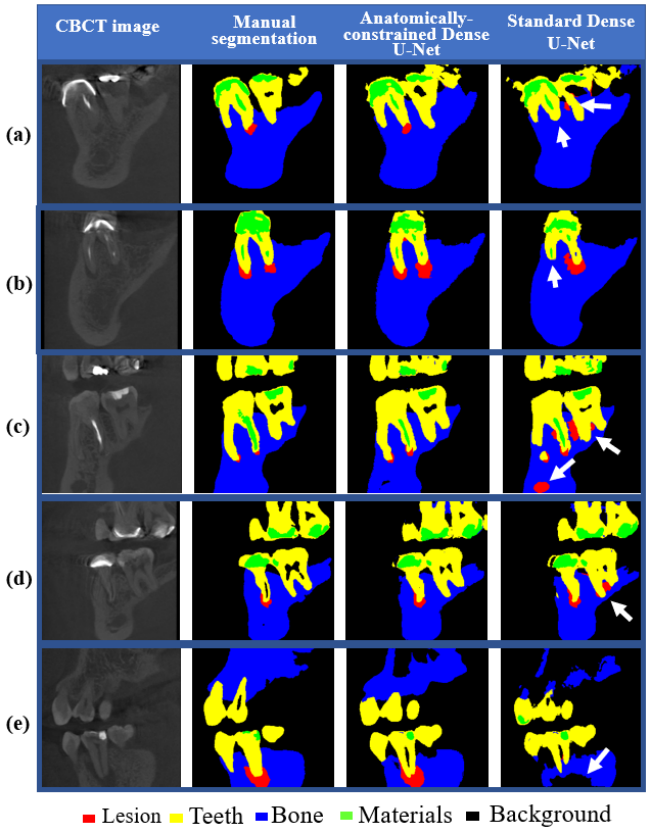


Fig. 3. CBCT images, manual segmentation, and segmentations by two competing algorithms for (a)–(e) five example cases. White arrows on the images of Dense U-Net (last column) point to areas where the proposed anatomically-constrained Dense U-Net outperforms.

Fig. 2 shows the average curves over the four iterations of the fourfold cross-validation. We compared the learning curves of the anatomically constrained Dense U-Net and Dense U-Net under the focal loss. The learning curves under other losses can be generated in a similar way but skipped due to the space limit. From Fig. 2, we can draw several observations: First, no overfitting of the proposed algorithm is observed as the validation curve keeps growing, and the gap between the validation and training curves is small. Second, we can see a clear improvement in the validation curve of the proposed algorithm at the 50th epoch after the proposed anatomical constraint-based regularization takes effect, compared with Dense U-Net.

Among the example segmentation results in Fig. 3, we can see that in Fig. 3(a), Dense U-Net cannot identify the lesion and produce a false alarm. Some pixels around the lesion are labeled as "background" (black in color) by Dense U-Net. The false lesion has contact with "background," which is prohibited by our proposed algorithm. In Fig. 3(b), Dense U-Net misses one lesion; also, it fails to detect a piece of materials (green). Our algorithm performs much better in these regards. In Fig. 3(c), the lesions are very small. Some large false lesions are reported by Dense U-Net. The lesions are captured by the proposed algorithm correctly. In Fig. 3(d), Dense U-Net detects a lesion at a wrong root, while the proposed algorithm detects only the correct lesion. In Fig. 3(e), Dense U-Net misses the lesion at the edge of the image. Our algorithm can detect the correct shape of the lesion and bones.

V. CONCLUSION

Dental care automation is an important area where AI or deep learning can make a great contribution, especially in dental practices that use radiographic imaging. CBCT is an important imaging modality that is experiencing ever-growing utilization in various dental fields. However, little work has been done for developing AI or deep learning capabilities for CBCT segmentation and lesion detection. In this article,

we proposed a unified framework to combine oral-anatomical knowledge into the deep learning design. We showed that the proposed algorithm outperformed the standard Dense U-Net in both lesion detection precision and DICE indices. Our algorithm performed well with data from only 20 patients included in the training.

There are several directions we would like to pursue in future research. First, while our proposed framework has the capability of including unlabeled images in training, we did not explore this capability in this study. An immediate next step is to compare the performance of the current algorithm with an extended version of the algorithm that incorporates unlabeled images. This will extend our algorithm into a semisupervised learning algorithm. Second, we would like to develop mathematical encoding to account for other types of anatomical knowledge such as size, shape, and so on. Third, we would like to extend the current algorithm that is based on selected slices of each CBCT volume to a 3-D algorithm that can take the volumes as input. Last but not least, manual segmentation results have been used as the ground truth to train the deep learning algorithm. There could be human errors. Developing algorithms robust to manual segmentation errors could be a future direction.

Appendix: Proof of Proposition 1

We have the Lagrangian objective function as

$$\begin{aligned} &L(q,\lambda,\gamma)\\ &\quad=\mathrm{KL}(q(\mathbf{y}|\mathbf{x})||p_\theta(\mathbf{y}|\mathbf{x}))\\ &\qquad-E_q\left(\sum_{j=1}^{P}\sum_{k\in\Omega}\lambda_k f_j^{(k)}\left(y_j,\mathbf{y}_{\mathrm{NE}(j)}\right)\right)\\ &\qquad+\sum_j\gamma_j\left(\sum_{y_\mathbf{j}}q\left(y_j|\mathbf{x}\right)-1\right). \end{aligned}\tag{13}$$

For the KL divergence in the objective function, we can separate it to two parts with one part including q_j and another part including q_{-j}, where $q_{-j}=\prod_{i\neq j}q_i$. That is, the KL divergence can be written as

$$\begin{aligned} &\mathrm{KL}(q(\boldsymbol{y}|\boldsymbol{x})||p_\theta(\boldsymbol{y}|\boldsymbol{x}))\\ &=E_q\left[\log q(\boldsymbol{y}\,|\,\boldsymbol{x})\right]-E_q\left[\log p_\theta(\boldsymbol{y}\,|\,\boldsymbol{x})\right]\\ &=E_{q_j}\left[\log q_j\left(y_j\,|\,\boldsymbol{x}\right)\right]+E_{q_{-j}}\left[\log q_{-j}\left(\boldsymbol{y}_{-j}\,|\,\boldsymbol{x}\right)\right]\\ &\quad-E_{q_j}\left[\log p_\theta\left(y_j\,|\,\boldsymbol{x}\right)\right]-E_{q_{-j}}\left[\log p_\theta\left(\mathbf{y}_{-j}\,|\,\boldsymbol{x}\right)\right]\\ &=\mathrm{KL}\left(q_j\left(y_j|\mathbf{x}\right)||p_\theta\left(y_j|\mathbf{x}\right)\right)+\mathrm{KL}\left(q_{-j}\left(\boldsymbol{y}_{-j}|\mathbf{x}\right)||p_\theta\left(\boldsymbol{y}_{-j}|\mathbf{x}\right)\right). \end{aligned}\tag{14}$$

For the expectation part in the objective function, we can separate the joint expectation over the entire image into three parts

$$\begin{aligned} &E_q\left[\sum_{j=1}^{P}\sum_{k\in\Omega}\lambda_k f_j^{(k)}\left(y_j,\mathbf{y}_{\mathrm{NE}(j)}\right)\right]\\ &\quad=E_{q_{i\notin NE(j),j}}\left[\sum_i\sum_{k\in\Omega}\lambda_k f_i^{(k)}\left(y_i,\mathbf{y}_{\mathrm{NE}(i)}\right)\right]\\ &\qquad+E_{q_{i\in\mathrm{NE}^2(j)}}\left[\sum_i\sum_{k\in\Omega}\lambda_k f_{i\in NE(j)}^{(k)}\left(y_i,\mathbf{y}_{\mathrm{NE}(i)}\right)\right]\\ &\qquad+E_{q_j}\left[E_{q_{-j}}\left[\sum_{k\in\Omega}\lambda_k f_j^{(k)}\left(y_j,\mathbf{y}_{\mathrm{NE}(j)}\right)\right]\right]. \end{aligned}\tag{15}$$

Here, $\mathrm{NE}^2(i)=\mathrm{NE}(\mathrm{NE}(i))\cup\mathrm{NE}(i)$. The third term in (15) can be further simplified as

$$E_{q_{-j}}\left[\sum_{k\in\Omega}\lambda_k f_j^{(k)}\left(y_j,\mathbf{y}_{\mathrm{NE}(j)}\right)\right]=\sum_{k\in\Omega}\lambda_k g_j^{(k)}\left(y_j\right).\tag{16}$$

In (15), the first term will be zero after taking the derivation over q_j. The second term will be

$$\begin{aligned} &E_{q_{i\in\mathrm{NE}^2(j)}}\left[\sum_i\sum_{k\in\Omega}\lambda_k f_{i\in NE(j)}^{(k)}\left(y_i,\mathbf{y}_{\mathrm{NE}(i)}\right)\right]\\ &\quad=\sum_i\sum_{k\in\Omega}\lambda_k E_{q_{i\in\mathrm{NE}^2(j)}}\left[f_{i\in NE(j)}^{(k)}\left(y_i,\mathbf{y}_{\mathrm{NE}(i)}\right)\right] \end{aligned}\tag{17}$$

$$\begin{aligned} &E_{q_{i\in\mathrm{NE}^2(j)}}\left[f_{i\in NE(j)}^{(k)}\left(y_i,\mathbf{y}_{\mathrm{NE}(i)}\right)\right]\\ &\quad=\sum_{p\in NE(j)}\left\{\begin{array}{l}q_p\left(y_p\neq k|\mathbf{x}\right)+\\ q_p\left(y_p=k|\mathbf{x}\right)\times\prod_{i\in NE(p)}q_i\left(y_i\neq k'|\mathbf{x}\right)\end{array}\right\}\\ &\frac{\partial E_{q_{i\in\mathrm{NE}^2(j)}}\left[f_{i\in NE(j)}^{(k)}\left(y_i,\mathbf{y}_{\mathrm{NE}(i)}\right)\right]}{\partial q_j}\\ &\quad=\sum_{p\in NE(j)}\left[\begin{array}{l}q_p\left(y_p=k|\mathbf{x}\right)\times\\ \prod_{i\in NE(p),\mathrm{i}\neq\mathrm{j}}q_i\left(y_i\neq k'|\mathbf{x}\right)\end{array}\right]\ll 1. \end{aligned}\tag{18}$$

Because this second-order term will always be product of four terms lower than one, which make it considerably lower than one, while $g_j\left(y_j\right)$ can be one in many cases. Therefore, when updating q_j, we ignore the second-order term.

Then, we can calculate partial gradient of objective function as

$$\begin{aligned} \frac{\partial L(q,\lambda,\gamma)}{\partial q_j}&=\log q_j\left(y_j|\mathbf{x}\right)+1-\log p_\theta\left(\mathrm{y_j}|\mathbf{x}\right)\\ &\quad-\sum_{k\in\Omega}\lambda_k g_j^{(k)}\left(y_j\right)+\gamma_j=0 \end{aligned}\tag{19}$$

$$q_j^*\left(\mathrm{y_j}|\mathbf{x}\right)=\frac{p_\theta\left(\mathrm{y_j}|\mathbf{x}\right)\exp\left(\sum_{k\in\Omega}\lambda_k g_j^{(k)}\left(y_j\right)\right)}{\exp\left(\gamma_j+1\right)}\tag{20}$$

$$\begin{aligned} \frac{\partial L(q,\lambda,\gamma)}{\partial\gamma_j}&=\sum_{y_\mathbf{j}}q\left(y_j|\mathbf{x}\right)-1\\ &=\sum_{y_\mathbf{j}}\frac{p_\theta\left(\mathrm{y_j}|\mathbf{x}\right)\exp\left(\sum_{k\in\Omega}\lambda_k g_j^{(k)}\left(y_j\right)\right)}{\exp\left(\gamma_j+1\right)}-1=0. \end{aligned}\tag{21}$$

Therefore, from (21), the normalizing constant of $q\left(y_j|\mathbf{x}\right)$ can be obtained as

$$c_j=\exp\left(\gamma_j+1\right)=\sum_{y_\mathbf{j}}p_\theta\left(\mathrm{y_j}|\mathbf{x}\right)\exp\left(\sum_{k\in\Omega}\lambda_k g_j^{(k)}\left(y_j\right)\right).\tag{22}$$

REFERENCES

[1] K. L. Dutra *et al.*, "Diagnostic accuracy of cone-beam computed tomography and conventional radiography on apical periodontitis: A systematic review and meta-analysis," *J. Endodontics*, vol. 42, no. 3, pp. 356–364, Mar. 2016.

[2] J. M. Parker, A. Mol, E. M. Rivera, and P. Z. Tawil, "Cone-beam computed tomography uses in clinical endodontics: Observer variability in detecting periapical lesions," *J. Endodontics*, vol. 43, no. 2, pp. 184–187, Feb. 2017.

[3] A. Aamodt *et al.*, "Determination of the hounsfield value for CT-based design of custom femoral stems," *J. Bone Joint Surg. Brit.*, vol. 81-B, no. 1, pp. 143–147, Jan. 1999.

[4] S. Prevrhal, K. Engelke, and W. A. Kalender, "Accuracy limits for the determination of cortical width and density: The influence of object size and CT imaging parameters," *Phys. Med. Biol.*, vol. 44, no. 3, p. 751, 1999.

[5] F. Abdolali, R. A. Zoroofi, Y. Otake, and Y. Sato, "Automatic segmentation of maxillofacial cysts in cone beam CT images," *Comput. Biol. Med.*, vol. 72, pp. 108–119, May 2016.

[6] S. Li, T. Fevens, A. Krzyżak, C. Jin, and S. Li, "Toward automatic computer aided dental X-ray analysis using level set method," in *Proc. Int. Conf. Med. Image Comput.-Assist. Intervent*, 2005, pp. 670–678.

[7] K. Okada, S. Rysavy, A. Flores, and M. G. Linguraru, "Noninvasive differential diagnosis of dental periapical lesions in cone-beam CT scans," *Med. Phys.*, vol. 42, no. 4, pp. 1653–1665, Mar. 2015.

[8] L. T. Hiew, S. H. Ong, K. W. C. Foong, and C. Weng, "Tooth segmentation from cone-beam CT using graph cut," in *Proc. 2nd APSIPA Annu. Summit Conf.*, 2010, pp. 272–275.

[9] E. D. Berdouses, G. D. Koutsouri, E. E. Tripoliti, G. K. Matsopoulos, C. J. Oulis, and D. I. Fotiadis, "A computer-aided automated methodology for the detection and classification of occlusal caries from photographic color images," *Comput. Biol. Med.*, vol. 62, pp. 119–135, Jul. 2015.

[10] S. Kida *et al.*, "Cone beam computed tomography image quality improvement using a deep convolutional neural network," *Cureus*, vol. 10, no. 4, p. e2548, 2018, doi: 10.7759/cureus.2548.

[11] Y. Lei *et al.*, "Image quality improvement in cone-beam CT using deep learning," *Proc. SPIE*, vol. 10948, Mar. 2019, Art. no. 1094827.

[12] H. Yang, K. Liang, L. Zhang, K. Kang, and Y. Xing, "Improve 3D cone-beam CT reconstruction by slice-wise deep learning," in *Proc. IEEE Nucl. Sci. Symp. Med. Imag. Conf. (NSS/MIC)*, Nov. 2018, pp. 1–3.

[13] Y. Pei, X. Ai, H. Zha, T. Xu, and G. Ma, "3D exemplar-based random walks for tooth segmentation from cone-beam computed tomography images," *Med. Phys.*, vol. 43, no. 9, pp. 5040–5050, Aug. 2016.

[14] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *Proc. Int. Conf. Artif. Neural Netw. Mach. Learn. (ICANN)*. Cham, Switzerland: Springer, 2018, pp. 270–279.

[15] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," 2017, *arXiv:1712.04621*. [Online]. Available: http://arxiv.org/abs/1712.04621

[16] S. Wang *et al.*, "Diabetic retinopathy diagnosis using multichannel generative adversarial network with semisupervision," *IEEE Trans. Autom. Sci. Eng.*, early access, Apr. 9, 2020, doi: 10.1109/TASE.2020.2981637.

[17] S. Jégou, M. Drozdzal, D. Vazquez, A. Romero, and Y. Bengio, "The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jul. 2017, pp. 11–19.

[18] K. M. Hermann *et al.*, "Teaching machines to read and comprehend," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 1693–1701.

[19] A. Kumar *et al.*, "Ask me anything: Dynamic memory networks for natural language processing," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1378–1387.

[20] H. Chen, M. Sun, C. Tu, Y. Lin, and Z. Liu, "Neural sentiment classification with user and product attention," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2016, pp. 1650–1659.

[21] Y. Shen, P.-S. Huang, J. Gao, and W. Chen, "ReasoNet: Learning to stop reading in machine comprehension," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2017, pp. 1047–1055.

[22] A. Sordoni, P. Bachman, A. Trischler, and Y. Bengio, "Iterative alternating neural attention for machine reading," 2016, *arXiv:1606.02245*. [Online]. Available: http://arxiv.org/abs/1606.02245

[23] B. Dhingra, H. Liu, Z. Yang, W. W. Cohen, and R. Salakhutdinov, "Gated-attention readers for text comprehension," 2016, *arXiv:1606.01549*. [Online]. Available: http://arxiv.org/abs/1606.01549

[24] H. Sun, B. Dhingra, M. Zaheer, K. Mazaitis, R. Salakhutdinov, and W. W. Cohen, "Open domain question answering using early fusion of knowledge bases and text," 2018, *arXiv:1809.00782*. [Online]. Available: http://arxiv.org/abs/1809.00782

[25] J. Sirignano and K. Spiliopoulos, "DGM: A deep learning algorithm for solving partial differential equations," *J. Comput. Phys.*, vol. 375, pp. 1339–1364, 2018.

[26] Z. Long, Y. Lu, X. Ma, and B. Dong, "PDE-net: Learning PDEs from data," 2017, *arXiv:1710.09668*. [Online]. Available: http://arxiv.org/abs/1710.09668

[27] M. Richardson and P. Domingos, "Markov logic networks," *Mach. Learn.*, vol. 62, nos. 1–2, pp. 107–136, 2006.

[28] A. D. A. Garcez *et al.*, "Neural-symbolic learning and reasoning: Contributions and challenges," presented at the AAAI Spring Symp. Ser. Stanford, CA, USA: Stanford Univ., Mar. 2015.

[29] G. G. Towell and J. W. Shavlik, "Knowledge-based artificial neural networks," *Artif. Intell.*, vol. 70, nos. 1–2, pp. 119–165, Oct. 1994.

[30] M. V. M. França, G. Zaverucha, and A. S. d'Avila Garcez, "Fast relational learning using bottom clause propositionalization with artificial neural networks," *Mach. Learn.*, vol. 94, no. 1, pp. 81–104, Jan. 2014.

[31] M. J. Kusner, B. Paige, and J. M. Hernández-Lobato, "Grammar variational autoencoder," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, 2017, pp. 1945–1954.

[32] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *J. Mach. Learn. Res.*, vol. 12, pp. 2493–2537, Aug. 2011.

[33] T. Karaletsos, S. Belongie, and G. Rätsch, "Bayesian representation learning with oracle constraints," 2015, *arXiv:1506.05011*. [Online]. Available: http://arxiv.org/abs/1506.05011

[34] Z. Hu, X. Ma, Z. Liu, E. Hovy, and E. Xing, "Harnessing deep neural networks with logic rules," 2016, *arXiv:1603.06318*. [Online]. Available: http://arxiv.org/abs/1603.06318

[35] Z. Hu *et al.*, "Deep generative models with learnable knowledge constraints," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 10501–10512.

[36] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput.-Assist. Intervent. (MICCAI)*. Cham, Switzerland: Springer, 2015, pp. 234–241.

[37] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4700–4708.

[38] A. Guha Roy, S. Conjeti, N. Navab, and C. Wachinger, "QuickNAT: A fully convolutional network for quick and accurate segmentation of neuroanatomy," *NeuroImage*, vol. 186, pp. 713–727, Feb. 2019.

[39] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1520–1528.

[40] N. Ibtehaz and M. S. Rahman, "MultiResUNet: Rethinking the U-Net architecture for multimodal biomedical image segmentation," *Neural Netw.*, vol. 121, pp. 74–87, Jan. 2020.

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.

[42] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "Unet++: A nested u-net architecture for medical image segmentation," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support.* Cham, Switzerland: Springer, 2018, pp. 3–11.

[43] Y. Miki *et al.*, "Classification of teeth in cone-beam CT using deep convolutional neural network," *Comput. Biol. Med.*, vol. 80, pp. 24–29, Jan. 2017.

[44] I.-B. Pavaloiu *et al.*, "Neural network based edge detection for CBCT segmentation," in *Proc. E-Health Bioeng. Conf. (EHB)*, Nov. 2015, pp. 1–4.

[45] A. Zakirov, M. Ezhov, M. Gusarev, V. Alexandrovsky, and E. Shumilov, "Dental pathology detection in 3D cone-beam CT," Diagnocat Co., Moscow, Russia, 2018, *arXiv:1810.10309*. [Online]. Available: https://arxiv.org/abs/1810.10309

[46] M. J. Wainwright and M. I. Jordan, "Graphical models, exponential families, and variational inference," *Found. Trends Mach. Learn.*, vol. 1, nos. 1–2, pp. 1–305, 2007.

[47] E. P. Xing, M. I. Jordan, and S. Russell, "A generalized mean field algorithm for variational inference in exponential families," 2012, *arXiv:1212.2512*. [Online]. Available: http://arxiv.org/abs/1212.2512

[48] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Oct. 2017, pp. 2980–2988.
[49] L. R. Dice, "Measures of the amount of ecologic association between species," *Ecology*, vol. 26, no. 3, pp. 297–302, 1945.
[50] M. Abadi *et al.*, "TensorFlow: A system for large-scale machine learning," in *Proc. 12th USENIX Symp. Operating Syst. Design Implement. (OSDI)*, 2016, pp. 265–283.
[51] T. Schloss, D. Sonntag, M. R. Kohli, and F. C. Setzer, "A comparison of 2- and 3-dimensional healing assessment after endodontic surgery using cone-beam computed tomographic volumes or periapical radiographs," *J. Endodontics*, vol. 43, no. 7, pp. 1072–1079, Jul. 2017.
[52] H. Ma *et al.*, "Volumetric assessment of sinus membrane dimensions in relation to endodontically treated and healthy teeth," *J. Endodontics*, vol. 44, p. e21, Mar. 2018.
[53] A. Poly *et al.*, "The ability of four different instrumentation systems in shaping oval canals: A micro-computed tomographic analysis," *J. Endodontics*, vol. 44, p. e40, Mar. 2018.

Zhiyang Zheng received the B.S. degree in statistics from the University of Science and Technology of China, Hefei, China, in 2018. He is currently pursuing the Ph.D. degree with the H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, USA.

He is currently a Research Assistant with the Georgia Institute of Technology, supervised by Dr. J. Li. His research interest includes medical image-based machine learning and deep learning.

Hao Yan received the B.S. degree in physics from Peking University, Beijing, China, in 2011, and the M.S. degree in statistics, the M.S. degree in computational science and engineering, and the Ph.D. degree in industrial engineering from the Georgia Institute of Technology, Atlanta, GA, USA, in 2015, 2016, and 2017, respectively.

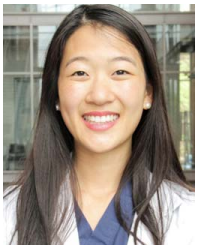
He is currently an Assistant Professor with the School of Computing, Informatics, and Decision Systems Engineering (SCIDSE), Arizona State University (ASU), Tempe, AZ, USA. His research interests focus on developing scalable statistical learning algorithms for large-scale high-dimensional data with complex heterogeneous structures to extract useful information for the purpose of system performance assessment, anomaly detection, intelligent sampling, and decision making.

Dr. Yan was a recipient of multiple awards including the Best Paper Award in the IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING (TASE) and the ASQ Brumbaugh Award. He is a member of INFORMS and IIE.

Frank C. Setzer received the Doctor of Dental Surgery (D.D.S.) and Doctor of Medicine in Dentistry (D.M.D.) degrees in dentistry from Friedrich–Alexander University Erlangen–Nuremberg, Erlangen, Germany, in 1995 and 1998, respectively, and the Endodontic Specialty Certificate, the M.S. degree in oral biology, and the D.M.D. degree from the University of Pennsylvania, Philadelphia, PA, USA, in 2006, 2008, and 2010, respectively.

He is currently an Assistant Professor of endodontics with the University of Pennsylvania. His research interests are the clinical detection, prognosis, and healing of apical periodontitis utilizing cone beam computed tomography (CBCT) imaging in endodontics.

Dr. Setzer is a Dental Faculty Member of OKU, a member of the American Association of Endodontists AAE, and a Diplomate of the American Board of Endodontics ABE.



Katherine J. Shi received the B.A. degree in molecular and cell biology from the University of California at Berkeley, Berkeley, CA, USA, in 2015, and the Doctor of Medicine (D.M.D.) degree in dentistry from the School of Dental Medicine, University of Pennsylvania, Philadelphia, PA, USA, in 2019.

She is currently pursuing the Endodontics residency with the School of Dental Medicine, Tufts University, Boston, MA, USA. Her research interests are on the intersections of radiology and endodontics.

Mel Mupparapu received the D.M.D. degree from the School of Dental Medicine, University of Pennsylvania, Philadelphia, PA, USA, in 1996.

He worked as a Faculty Member with Penn Dental Medicine, Philadelphia, from 1996 to 2002. He joined the Rutgers School of Dental Medicine, Newark, NJ, USA, in 2002, as the Director of Radiology, where he became a Full Professor in 2005 and continued there until 2011. His current interests include research related to cone beam computed tomography (CBCT), artificial intelligence (AI), and remote dental consultation. He returned to Penn Dental Medicine, in 2011, as the Director of Radiology. He has over 130 peer-reviewed publications, numerous technology-related grants, and invited presentations to his credit.

Dr. Mupparapu became board certified in oral and maxillofacial radiology in 1999. He was inducted to the Master Educators Guild in 2007 and received the U.S. State Department Sponsored Fulbright Scholarship from the University of Malta in 2009.

Jing Li (Member, IEEE) received the B.S. degree from Tsinghua University, Beijing, China, in 2007, and the M.A. degree in statistics and the Ph.D. degree in industrial and operations engineering from the University of Michigan, Ann Arbor, MI, USA, in 2005 and 2007, respectively.

She is currently a Professor with the H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA, USA. Her research interests are statistical modeling and machine learning for healthcare applications.

Dr. Li was a recipient of the NSF CAREER Award. She is a member of IISE and INFORMS.