Automated Feature Detection in Dental Periapical Radiographs using Deep Learning

Abstract

Objectives: To investigate automated feature detection, segmentation, and quantification of common periapical findings in periapical radiographs (PAs) using deep learning (DL)-based computer vision techniques.

Methods: Caries, alveolar bone recession, and interradicular radiolucencies were labelled on 206 digital PAs by 3 specialist clinicians (2 oral pathologists and an endodontist). This dataset was divided into 'Training and Validation' and 'Test' datasets consisting of 176 and 30 PAs, respectively. Multiple transformations of image data were used as input to deep neural networks during training. Outcomes of existing and purpose-built DL architectures were compared to identify the most suitable architecture for automated analysis.

Results: The U-Net architecture and its variant outperformed other DL algorithms in all performance metrics. The overall best performing architecture on the validation dataset was 'U-Net+Densenet121' (mIoU = 0.501, Dice coefficient = 0.569). Performance of all architectures degraded on the 'Test' dataset; 'U-Net' delivered the best performance (mIoU = 0.402, Dice coefficient = 0.453). Interradicular radiolucencies were the most difficult to segment.

Conclusions: DL has potential for automated analysis of PAs but warrants further research. Among existing, off-the-shelf, architectures, U-Net and its variants delivered the best performance. Further performance gains can be obtained via purpose-built architectures and a larger multi-centric cohort.

Keywords: Medical Image Segmentation, Caries, Bone Recession, Interradicular Radiolucency, Deep Learning, Dental Radiography, Artificial Intelligence.

Introduction:

Digital radiographs are routinely employed by dentists to assess the extent of caries; examine root morphology; evaluate status of alveolar bone; determine the need for orthodontic treatment; and evaluate dental, jaw and sinus diseases[1-4]. Common radiographs used in clinical practice include periapicals, bitewings, and orthopantomograms (OPT) [5].

Periapical radiographs (PAs) are a very commonly used in intraoral radiography. They provide localized information on the presence and extent of caries, restorations, interradicular radiolucencies, root and root canal morphology, the length and adequacy of endodontic obturation, the level of alveolar bone, and the periodontal ligament space. Although all dentists are well trained in interpreting these images, factors such as variation in contrast, angulation, and magnification can result in faulty diagnoses. Other factors that can influence interpretation include the experience and knowledge of the dentist as well as fatigue during the examination of radiographs[6].

Furthermore, interpretation of conventional radiographs is subjective and creates the potential for inconsistencies between dentists [7, 8]. Despite this limitation, the easy accessibility and clinical reliability of PAs make them a preferred choice for diagnosing common dental problems [9].

These challenges make the use of automated and more objective analysis an attractive option for aiding in diagnosis and improving patient care. Deep learning (DL) encompasses a set of techniques inspired from the anatomy of the brain that have become quite popular in artificial intelligence and computer vision. These techniques have improved our ability to build software for automated analysis and evaluation of images with widespread application in medical image analysis. Recent advances in DL have shown the potential for automated identification and quantification of radiological and pathological features to improve consistency of diagnosis and standardization of care as well as provide quantifiable outcomes [10, 11]. However, application of DL in dental radiology remains poorly explored.

There have been limited attempts at automated analysis of dental radiographs using DL with reported studies mostly exploring caries detection and tooth identification, with no attempt at shape segmentation that would guide treatment [12-15]. Furthermore, the accuracy of reported detection has been variable and somewhat suboptimal, highlighting the need for further research in this area.

The objective of the research was to compare the diagnostic efficacy of 4 segmentation architectures in computer-based deep learning in the diagnosis of caries, alveolar bone recession (ABR), and interradicular radiolucencies (IRR). The null hypothesis stated that there would be no significant differences between the 4 architectures in diagnosing these abnormalities.

Materials and Methods

Dataset

The initial data used for training and validation in this study contained PAs collected from a single dental practice over a 6-month period between January and July 2019. This 'Training and Validation' dataset was selected out of an original total of 200 periapical radiographs and comprised 176 PAs that contained 135 instances of caries, 149 instances of alveolar bone recession (ABR), and 57 instances of interradicular radiolucency (IRR). Using data from only a single source is not ideal because the performance of AI algorithms tends to degrade when tested on data from sources to which they have not been exposed before. This degradation can be due to factors such as variation in the physical properties of data acquisition devices/instruments at different sources. Consequently, we also evaluated the performance of our approach on a smaller 'Testing' dataset of 30 PAs collected from 2 more dental practices that were different from the one that provided the initial 'Training and Validation' data. For both datasets, diagnostically acceptable PAs acquired by using a standard paralleling technique were selected. The radiographs were anonymized by the source practices prior to being shared with the research team.

Step 1 in our protocol was data labeling. Data on the 'Training and Validation" radiographs was labeled by 3 experienced clinicians including an American board certified oral pathologist (AK), a specialist in endodontics (MM) with extensive experience in diagnosis

and interpretation of dental radiology, and a consultant specialist in oral and maxillofacial pathology from the UK (SAK) with expertise in both oral and maxillofacial radiology and surgery. During the labeling process one examiner (AK) meticulously annotated caries, ABR, and IRR, ensuring that the shape of the annotated region overlapped with the boundary of the underlying region of interest. Three colors were employed to label the three distinct regions of interest; red was used to label caries, blue for ABR, and green for IRR. The remaining two examiners (MM and SAK) examined the labels drawn by the first examiner and accepted or rejected them. Radiographs on which at least 2 out of 3 examiners did not agree were excluded from the dataset. Out of the 200 original radiographs, 176 were retained in the 'Training and Validation' set, while from 31 additional radiographs, 30 were selected for the 'Testing' data set, as listed above.

The data labeling process is illustrated in Figure 1. The output of this process consisted of the color-coded labeled images ('Reference' label masks) of exactly the same size as the input periapical images indicating the shape of the three features of interest (caries, ABR, and IRR). During training, the reference label masks were used to locate and learn the visually distinct characteristics of each feature of interest. A trained network was able to take an unseen image and output a 'Predicted' label mask containing the (estimated) shape of any caries, ABR, and IRR in the image. Prediction performance was evaluated by comparing the Predicted label mask with the corresponding Reference label mask.

Step 2 in our approach was training and validation, in which the labeled dataset was first augmented (see 'Data Augmentation' below) and then employed to train and evaluate the performance of different DL architectures. Given the relatively small size of the dataset, we employed 4-fold cross-validation to measure the performance of different neural network architectures in this step. More specifically, the training and validation dataset was partitioned into 4 sets of approximately equal size. At any one time, 3 out of these 4 sets were used for training whereas the 4th set was used for performance evaluation or testing. This process was repeated until performance had been evaluated on each of the 4 partitions of the training and validation dataset. A graphic illustration of 4-fold cross-validation is provided in Figure 2.

Finally, during step 3 (testing), the entire validation dataset was used to train the network and performance was evaluated on the unseen test dataset. As explained earlier, this was

done to gauge the degradation in performance usually seen when a trained network is exposed to data from unfamiliar sources.

Data Augmentation

Data augmentation is a common pre-processing technique employed prior to feeding data samples to a neural network during training. It entails increasing the number of cases (or features) in the original dataset set by applying realistic transformations (i.e., mock computer-generated images) that are representative of variations expected to occur in real life. For example, the same radiograph may be flipped or rotated to generate multiple copies that could represent different viewing angles. Data augmentation also mitigates the adverse impact of class imbalance (the predominance of one feature) by generating a relatively large number of images for features with low prevalence in the original dataset. We experimented with different types of transformations and found that magnification, vertical flip, translation, rotation, horizontal flip, shear, crop, and elastic transformations were the most useful and delivered the largest gains in performance (when compared to non-augmented data). Sample images resulting from application of some of these transformations are shown in the data augmentation block of Figure 2.

Segmentation Architectures and Training

The primary objective of our algorithm was to assign a class label or identity (caries, ABR, IRR, or background) to every pixel of an input periapical radiographic image. In computer vision, this process of labeling all pixels in an image is known as semantic segmentation and a large number of available DL architectures can be employed for this purpose. We explored a few of the existing as well as some novel architectures for semantic segmentation of the 3 features that were of interest to us. Most deep neural network-based semantic segmentation algorithms employ an Encoder-Decoder architecture constructed by using convolutional neural networks (CNNs). Every layer of a CNN consists of a set of kernels or filters. A single kernel is a feature extractor that can be used to find the location(s) of a feature (or geometric shape) in an image. The presence of a feature in an image can be detected by first dividing it into small, equal-sized patches and then multiplying each patch with a kernel that is similar in size to the image patches. Patches containing features that

are similar in shape to the kernel result in high values whereas patches containing different shapes result in values close to zero.

This process of using a kernel as a template to search for shapes in images is known as 'convolution' and is illustrated in Figure 3. The output of a convolution operation is also an image; however, it is generally referred to as a 'feature map' since it highlights image patches that contain shapes similar to the kernel used, and filters out patches that are different. For example, feature map-1 in Figure 3 is obtained by convolution of the input image with a circular kernel (Kernel-1); it highlights regions containing only circular features. Similarly, Kernel-2 is cylindrical and convolving it with the image highlights regions containing cylindrical shapes.

The encoder of a typical segmentation architecture consists of successive blocks of CNNs that are employed to extract different shapes in the input image. For example, the U-Net segmentation architecture is shown in Figure 4, in which the number of feature maps in every block is indicated by the value written across it. The size of the image/feature maps input to any block is indicated by the value written below it (all images/feature maps are square in size, with an equal number of X and Y pixels). For example, the second block of the encoder comprises 128 feature maps which are obtained after application of convolution to feature maps of size (284 x 284) pixels that are input to it.

In general, the application of convolution reduces the size of images. Therefore, the size of feature maps decreases as we pass through blocks of the encoder. The first block of the encoder extracts simple geometric features (such as horizontal and vertical lines) from the input image. The subsequent layers learn to extract more complex features by combining simpler features input to them by the preceding layers. For example, lines and curves can be combined to construct shapes like polygons and circles that can be further combined to construct more complex shapes like objects. Consequently, the final block of the encoder consists of a large number of feature maps, each of which describes the approximate location of a complex shape or object within the image. However, successive application of convolution operations means that the size of the feature maps is significantly smaller than the size of the input image. A decoder is then applied to upsample the encoder feature

maps in a step-by-step manner using the up-convolution operation. In the U-Net architecture, each block of the decoder combines information it receives from its preceding decoder block with its corresponding peer block in the encoder. At every block of the decoder, the size of the feature maps is increased whereas the number of feature maps is halved. This is repeated until we are left with a single feature map that describes the shape and location of the objects/regions of interest in the original input image. During training, a network uses input images and reference label masks to learn kernels and other parameters of the network that enable it to output predicted label masks that are similar to the reference label masks. Once trained, the learned kernels and network parameters are used to generate predicted label masks for unseen images that are input to the network.

Semantic segmentation is being widely applied in computer vision and there are numerous architectures available for this purpose. For our experiments, we selected 4 neural network architectures, of which 3 were existing architectures (U-net [16], XNet [17] and SegNet [18] and 1 was a custom-built architecture constructed by replacing the encoder layer of U-net with the Densenet121 architecture[19]. Among the three existing architectures, U-Net and XNet were purpose-built for medical image segmentation. U-Net has a proven track record of delivering good results in medical imaging applications where training data is limited in size [20]. Periapical radiographs are X-ray images. Therefore, the XNet architecture, which was purpose-built for radiological image segmentation, was also selected for evaluation on our dataset. The third segmentation architecture used was SegNet, which is a popular architecture for segmentation of natural images. The primary purpose of including SegNet in our evaluation was to gauge the performance difference between architectures designed for natural images and architectures built specifically for medical/dental images. The fourth architecture was a variant of U-net and was constructed by replacing its encoder with a more recent encoder architecture, Densenet121 [18]. The fourth (custom-built) architecture was tested primarily because since U-net's inception in 2015 a number of new encoder architectures have been proposed. Therefore, substituting its encoder with a relatively recent encoder architecture could potentially deliver an improvement in performance.

All architectures employed were implemented using the Keras and Tensorflow frameworks, and trained using graphic processing unit (GPU) instances on the Amazon Web Services

cloud platform. The process was started by training original (unmodified) versions of U-net, XNet, and SegNet. Results demonstrated that U-Net delivered the best performance. Consequently, we experimented further with U-Net by replacing its encoder layers with other popular encoder architectures and investigating whether doing so resulted in additional performance gains. All images were resized to 256 × 256 pixels for standardization before being input to the network for training or testing.

3.4. Evaluation Metrics

Three distinct metrics were employed to evaluate the performance of different approaches. The first metric employed was the Intersection over Union (IoU) which is the ratio of the number of pixels that are in common (or overlap) between the reference label mask and the predicted label mask output by the network to the total number of pixels in both masks. The IoU is calculated using the following equation:

$$IoU = \frac{Ground\ Truth\ \cap\ Prediction}{Ground\ Truth\ U\ Prediction} = \frac{TP}{TP + FN + FP} \qquad (1)$$

TP denotes true positives and is the number of pixels that are correctly predicted as belonging to the target class. Similarly, FP and FN denote the number of false positive and false negative pixels, respectively. Performance was evaluated using the mean IoU (mIoU), which is the mean value of the individual IoU values observed on the test/validation images.

The second evaluation metric we employed was the Dice coefficient, which is defined as:

$$Dice = \frac{2TP}{2TP + FN + FP} \qquad (2)$$

While both the above metrics are quite similar, the IoU penalizes single instances of bad segmentation much more than the Dice coefficient. Consequently, an algorithm which is correct for the vast majority of instances but makes incorrect decisions in a few instances may result in an IoU that is much lower than the corresponding Dice coefficient, which is better at reflecting average performance and not overly sensitive to a few instances of bad performance. Just like the IoU, performance evaluation was conducted using the mean Dice coefficient, which is the mean value of the individual Dice coefficients observed on the

test/validation images. An ideal segmentation algorithm that perfectly matches the reference label maps will result in mIoU and Dice coefficient values of 1, whereas an algorithm that results in no overlap between reference and predicted label mask will generate mIoU and Dice coefficient values equal to 0.

Results

Performance evaluation of different architectures was done using two different approaches: (1) 4-fold cross-validation on the validation dataset and (2) testing on an independent test dataset collected from sources not included in the validation data, as described above.

Validation Dataset

The mIoU and Dice Coefficient values obtained for the validation dataset are shown in Table I. In 4-fold cross-validation the data was divided into four partitions which were then used as test sets one-by-one. Therefore, every value in Table I was obtained by averaging over the values observed for the 4 partitions of the dataset. It can be observed that for the 3 off-the-shelf architectures (U-Net, XNet, and SegNet), the best segmentation performance was obtained for the U-Net architecture (average mIoU = 0.466; average Dice coefficient = 0.534). The U-Net+Densenet121 architecture gave the overall best performance on the validation dataset with average mIoU and Dice coefficient values of 0.501 and 0.569 respectively. Among the three features studied, segmentation of ABR was the easiest to identify, with the highest mIoU = 0.440 obtained by the U-Net+Densenet121 architecture. An mIoU of 0.440 implies that, on average, there is 44% percent overlap between regions identified as ABR in reference and predicted label masks. Similarly, the Dice coefficient for ABR was also highest with U-Net+Densenet121 (0.556).

Segmentation of caries resulted in similar performance, generating an mIoU of 0.428 with U-Net+Densenet121 (Dice coefficient = 0.532). However, segmentation of IRR seemed more challenging, with mIoU = 0.173 and the Dice coefficient = 0.206. This was most likely due to the relatively small number of instances of IRR in the validation dataset compared with the other two features. Segmentation of background, which includes everything that is not a part of one of the 3 studied features, appeared to be easier and the performance metrics

were quite high. However, it is worth noting that most regions in a radiographic image can be a part of the background. Therefore, these numbers were somewhat biased by the high prevalence of the background class. Overall, the mIoU and Dice coefficient values exhibited similar trends regarding performance of the 4 architectures and relative ease of segmentation of the 3 disease conditions. On average, SegNet produced the poorest mIoU and Dice coefficient values.

Test Dataset

The performance metrics observed for the test dataset are presented in Table II. The networks were trained on the entire validation dataset and then tested on the unseen test dataset. Overall performance was worse as compared to the validation dataset. However, this was expected since most DL approaches exhibit performance degradation when tested on data from sources different from those in the training data. We could have improved the performance on the test dataset by mixing examples of all 3 sources in the validation and test datasets. We chose not to do so because we wanted to keep the testing conditions challenging and as close to real life deployment scenarios as possible. In terms of overall average performance, the best network architecture was U-Net instead of U-Net+Densenet121 for both mIoU and the Dice coefficient. The average mIoU for U-Net decreased from 0.466 (for the validation dataset) to 0.402 (for the test dataset), a degradation of 13.8%. For the Dice coefficient, there was a decrease from 0.534 to 0.453 (15.2%). U-Net+Densenet121 yielded the second-best performance on the test dataset with an average mIoU of 0.383, representing a degradation of 23.5% from the validation dataset value of 0.501. The Dice coefficient for the test dataset was 0.434 compared to 0.569 in the validation dataset, or a degradation of 23.7%. This could be due to the larger number of parameters in U-Net+DensetNet121 (25 million compared to 5 million for U-Net). Segmentation of individual features exhibited trends similar to those observed in the validation dataset for both mIoU and the Dice coefficient. ABR was the easiest to segment, followed by caries and IRR. Segmentation performance of IRR was quite low. However, the test dataset only had one radiograph which contained any instances of IRR. The low mIoU and Dice coefficient can be attributed to the low prevalence of IRR in the dataset and may not be truly reflective of algorithm performance. The SegNet architecture had the lowest average mIoU and Dice coefficient values.

In order to better understand the actual performance, a comparison of reference and predicted label masks of six images from the test dataset is presented in Figure 5, in which 56 images (A through F) are presented in columns (a) through (f), respectively. Due to the naturally low occurrence of IRR, the test dataset contained only a single image with IRR. This image and its corresponding reference and prediction label masks are shown in Figure 5 as image E. The other 5 images shown in Figure 5 were randomly selected from the test dataset. It can be observed that ABR was the easiest to segment and almost all network architectures did a reasonable job at this task. U-Net was the best performing architecture on the test dataset and was able to correctly detect the location of all 13 instances of ABR in the six sample images in Figure 5. However, the estimation of the shape of each instance of ABR was not perfect and could be improved further. Among U-Net predictions there were 5 instances of false positives, the largest of which can be observed in Image A. Segmentation of caries was more difficult. The performance of U-Net at this task returned the lower mIoU value of 0.166 and Dice coefficient of 0.202 (listed in Table II) compared with the values of 0.291 and 0.376, respectively, in the validation dataset, a degradation of 43.0% for mIoU and 46.3% for the Dice coefficient. Out of the 8 instances of caries in the images, U-Net was able to correctly locate only 3 in image A. Table II indicates that U-Net+Densenet121 was marginally better at segmenting caries (mIoU = 0.194, Dice coefficient = 0.239). The images in Figure 5 seem to corroborate this since U-Net+Densenet121 did not make any false predictions of caries in image A. Furthermore, visually it seemed that the shape of caries predicted by U-Net+Densenet121 (in image C and image F) were marginally better estimates than those produced by U-Net. Performance evaluation of IRR segmentation was challenging on the test dataset since it contained only a single image with IRR. However, visual inspection of the images indicated that U-Net outperformed the other architectures, giving the best estimate of the shape of the IRR in image E. It also gave the smallest number of false positives.

An independent t-test was employed to compare the mIoU values of U-Net with those of Xnet, SegNet, and U-Net+Densenet121. A p-value of less than 0.05 was considered significant. Significantly different results were obtained for U-Net vs. Xnet ($p = 0.006$) and U-Net vs. SegNet ($p < 0.002$). Despite U-Net+Densenet121 outperforming U-net in mIoU values, no statistically significant difference was noted between U-Net and U-Net+Densenet121 ($p = 0.198$).

Similarly, the Dice coefficients of U-Net were compared with Xnet, SegNet, and U-Net+Densenet121 using independent t-tests. These tests yielded significantly different values for comparisons between U-Net and Xnet ($p = 0.012$), and U-Net and SegNet ($p < 0.02$). However, no statistical difference was noted between the Dice coefficients of U-Net and U-Net+Densenet121 ($p=0.198$).

Discussion

Our findings show that DL has the potential to automatically detect the presence (detection) and shape (segmentation) of caries, ABR, and IRR in dental periapical radiographs. However, the performance evaluation metrics indicate that this is a challenging problem with significant room for improvement building upon existing work. Furthermore, performance degrades further when the algorithms are tested on data acquired from different sources. DL application to dental radiology has been limited and to the best of our knowledge there have been no prior attempts to segment ABR and IRR using these methods. Recently, deep CNNs were used to detect ABR on OPTs and cystic lesions on cone beam computed tomography scans. However, no attempt was made to segment shapes of the features of interest [14, 15]. In another study, different teeth were localized and classified using a faster R-CNN (where R-CNN stands for Region-based CNN) [21], but faster R-CNNs can only perform an estimation of the approximate shape and size of objects by putting rectangular bounding boxes around them.

The U-Net and XNet architectures were specifically designed for medical images and therefore outperformed the SegNet architecture, which was built primarily for natural images. Although the XNet architecture was designed for radiographic images, it was significantly outperformed by U-Net ($P = 0.006$ for mIoU, $P = 0.012$ for the Dice coefficient), which was somewhat unexpected. Both performance evaluation metrics, mIoU and the Dice coefficient, demonstrated similar trends. On the validation set, the overall highest average mIoU (0.501) was exhibited by the U-Net+Densenet121 architecture. This means that on average there was approximately 50% overlap between the corresponding features/classes (background, caries, ABR, and IRR) on predicted and reference label masks. Although seemingly low, these performance values cannot be dismissed outright for the following

reasons: (1) Semantic segmentation is a challenging task and algorithm performance varies widely depending on the complexity and size of the image dataset. For example, state-of-the-art segmentation algorithms have been shown to achieve mIoUs of around 0.80 on the Cityscapes dataset but degrade to 0.45 on the more challenging ADE20K dataset [22]. It is also worth highlighting that the Cityscapes and ADE20K datasets contain 5000 and 25,000 labelled images, respectively, which are significantly larger than the number of images in our datasets. (2) The highest average mIoU on the test dataset was seemingly low but visual inspection of the results in Figure 5 demonstrated that actual results were reasonable, as a value of > 0.5 is considered a good prediction on complex datasets of limited size. Although the estimation of feature shapes is not very precise, the best performing architecture (U-Net) was able to correctly locate a number of occurrences of caries, ABR, and IRR. Furthermore, it seems that it was also able to learn that caries is found in the coronal portion of teeth, ABR between teeth, and IRR around the roots. Therefore, although semantic segmentation in its current form cannot accurately estimate shapes of the features of interest, it could possibly be employed to highlight their approximate locations.

One of our limitations was that we used the interpretation of three experts as ground truth. While a consensus of all three examiners was required to accept the annotations, radiologic interpretation is subjective. Other limitations included a small size of our training and testing data, and acquisition of training radiographs from a single source.

In summary, our results are promising and acceptable but not outstanding. This can be attributed to two factors: (1) limited training data and (2) complexity of the segmentation task. To further improve, diversify, and clinically deploy our algorithms, we are currently working on extending our training dataset to include more radiographs from multiple sources. Furthermore, for the clinically relevant features assessed in our current study and for additional features (such as subtle tooth decay and periapical radiolucency), we plan to undertake research that will include determining diagnostic measures of accuracy such as sensitivity and specificity, plus performing receiver operating characteristic analyses to determine area under the curve (AUC) values as a measure of accuracy.

Conclusion

Findings from our pilot study show that DL can be a viable option for segmentation of caries, ABR, and IRR in dental radiographs. Our results demonstrated that a reasonable performance can be obtained by training existing deep neural networks provided that labelled training data is available. In terms of performance, the approaches based on the U-Net architecture and its variants delivered the best results. Furthermore, replacing the encoder layers of U-Net with other architectures also resulted in performance gains, in controlled settings. However, performance of the custom-built architecture degraded when tested on data from different sources. This sensitivity to data from varied sources was most likely due to the significant increase in the number of parameters when the smaller U-Net encoder was replaced with the larger Densenet121 encoder. Further research is required to conclusively establish whether replacing encoders can deliver noticeable performance gains.

Funding

University of Jeddah, Saudi Arabia (UJ-20-097-DR)

Amazon Web Services for their gift of $20,000.

References

[1] Keenan JR, Keenan AV. Accuracy of dental radiographs for caries detection. Evid Based Dent. 2016;17:43.

[2] Mardini S, Gohel A. Imaging of Odontogenic Infections. Radiol Clin North Am. 2018;56:31-44.

[3] Alimohammadi R. Imaging of Dentoalveolar and Jaw Trauma. Radiol Clin North Am. 2018;56:105-24.

[4] Van der Stelt PF. Panoramic radiographs in dental diagnostics. Ned Tijdschr Tandheelkd. 2016;123:181-7.

[5] Masthoff M, Gerwing M, Masthoff M, Timme M, Kleinheinz J, Berninger M, et al. Dental Imaging - A basic guide for the radiologist. Rofo. 2019;191:192-8.

[6] Singer M.K.S.H.R Challenges associated with digital radiology in dentistry. . EC Dent Sci. 2017;13:13-23.

[7] Molven O, Halse A, Fristad I. Long-term reliability and observer comparisons in the radiographic diagnosis of periapical disease. Intl Endod J. 2002;35:142-7.

[8] Sherwood IA. Pre-operative diagnostic radiograph interpretation by general dental practitioners for root canal treatment. Dentomaxillofac Radiol. 2012;41:43-54.

[9] Gupta A, Devi, P., Srivastava, R., & Jyoti, B. Intra oral periapical radiography - basics yet intrigue: A review. Bangladesh J Dent Res Edu. 2014;4:83-7.

[10] Thrall JH, Li X, Li Q, Cruz C, Do S, Dreyer K, et al. Artificial intelligence and machine learning in radiology: opportunities, challenges, pitfalls, and criteria for success. J Am Coll Radiol. 2018;15:504-8.

[11] Serag A, Ion-Margineanu A, Qureshi H, McMillan R, Saint Martin MJ, Diamond J, et al. Translational AI and deep Learning in diagnostic pathology. Front Med (Lausanne). 2019;6:185.

[12] Wang CW, Huang CT, Lee JH, Li CH, Chang SW, Siao MJ, et al. A benchmark for comparison of dental radiography analysis algorithms. Med Image Anal. 2016;31:63-76.

[13] Lee JH, Kim DH, Jeong SN, Choi SH. Detection and diagnosis of dental caries using a deep learning-based convolutional neural network algorithm. J Dent. 2018;77:106-11.

[14] Krois J, Ekert, T., Meinhold, L. et al. . Deep Learning for the radiographic detection of periodontal bone loss. Sci Rep. 2019;9:8495.

[15] Lee JH, Kim DH, Jeong SN. Diagnosis of cystic lesions using panoramic and cone beam computed tomographic images based on deep learning neural network. Oral Dis. 2020;26:152-8.

[16] Ronneberger O, Fischer, P., & Brox, T. . U-net: Convolutional networks for biomedical image segmentation. In: Navab Nassir and Horneggar, Joachim and Wells, William M. and Frangi, Alejandro F., editor. International conference on edical image computing and computer-assisted intervention. Munich, Germany: Springer 2015. p. 234-41.

[17] Bullock J, Cuesta-Lázaro, C., & Quera-Bofarull, A. XNet: A convolutional neural network (CNN) implementation for medical X-Ray image segmentation suitable for small datasets. In: Gimi B, Kroll A, editor. Medical Imaging 2019: Biomedical Applications in Molecular, Structural, and Functional Imaging. San Diego, California: SPIE; 2019. p. 109531Z.

[18] Badrinarayanan V, Kendall, A., & Cipolla, R. . Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans Pattern Anal Mach Intell. 2017;39:2481-95.

[19] Jégou S, Drozdzal, M., Vazquez, D., Romero, A., & Bengio, Y. . The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation. In: Rehg J, Liu Y, Wu Y, Taylor C, editor. IEEE conference on computer vision and pattern recognition workshops Hawaii, USA: IEEE; 2017. p. 11-9.

[20] Litjens G, Kooi T, Bejnordi BE, Setio AAA, Ciompi F, Ghafoorian M, et al. A survey on deep learning in medical image analysis. Med Image Anal. 2017;42:60-88.

[21] Chen H, Zhang K, Lyu P, Li H, Zhang L, Wu J, et al. A deep learning approach to automatic teeth detection and numbering based on object detection in dental periapical films. Sci Rep. 2019;9:3840.

[22] Z Huang XW, L Huang, C Huang, Y Wei, W Liu. Ccnet: Criss-cross attention for semantic segmentation. In: Kweon IS ,Paragios N, Yang M.H, Lazebnik S, editor. IEEE International Conference on Computer Vision. Seoul, South Korea: IEEE; 2019. p. 603-12.

Figure Legends

Figure 1: Illustration of the data labeling process. An unlabeled image was examined by the pathologists. The three regions of interest were highlighted using distinct color codes. The labeled image was the 'Reference' image mask that was extracted by the computer vision team for training the deep neural network.

Figure 2: Block diagram illustrating the various steps involved in building a deep learning based tool for automated analysis of dental pathoses. Step 2 involved training and validation, in which the labeled images ('Reference' masks) went through data augmentation, which involved changes in orientation of the images. The training and validation dataset was divided into 4 sets for 4-fold cross validation. At any one time, 3 out of these 4 sets were used for training and the 4th set was used for performance evaluation or testing. This process was repeated until performance had been evaluated on each of the 4 partitions of the training and validation dataset. In step 3 (testing), the entire validation

dataset was used to train the network and performance was evaluated on the unseen test dataset.

Figure 3: Illustration of how convolution can be used to extract features from images. The white shade in the feature maps indicates the presence of features that are similar to the kernel applied (kernel = computer program acting as a filter).

Figure 4: Block diagram of the U-Net architecture used for segmentation. The colored arrows represent convolution operations and activation functions. The number below each block indicates the x-y size of the block, e.g., the second block of the encoder is (284 x 284) pixels. The number along the side of each block represents the number of channels, e.g., the second block of the encoder contains 128 channels.

Figure 5: Sample comparisons of reference and predicted label masks of six images from the test dataset. The top row displays unlabeled images; the second row displays radiographs with reference label masks superimposed on top of them; rows 3 through 6 display the radiographs with predicted label masks of the 4 different architectures superimposed on top of them. Images in columns (a) through (f) are referred to as images A through F, respectively. All images except image E were picked randomly from the 'Test' dataset.

Figure 1

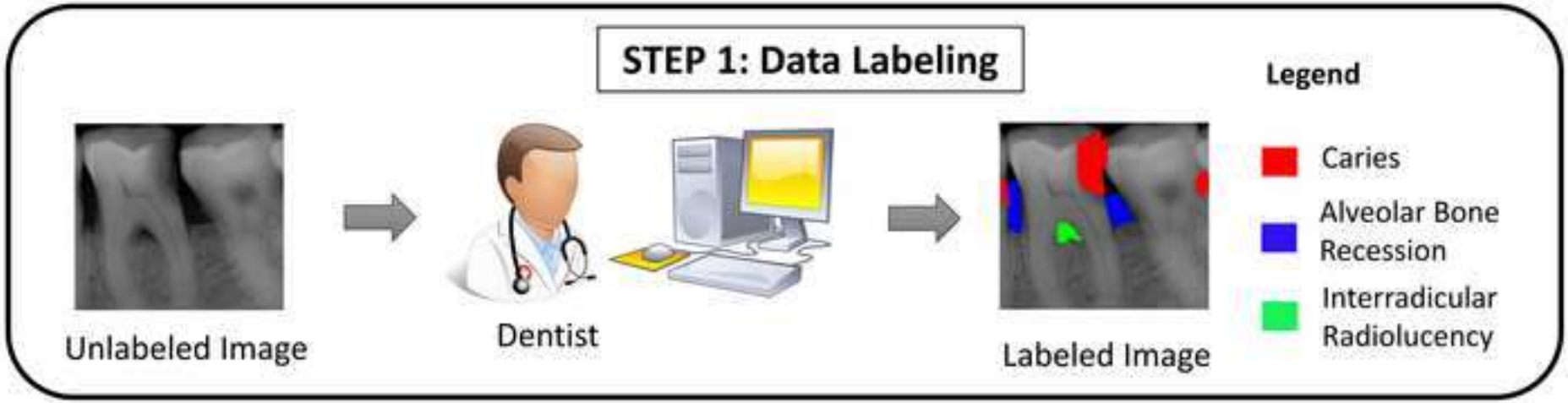


Figure 2

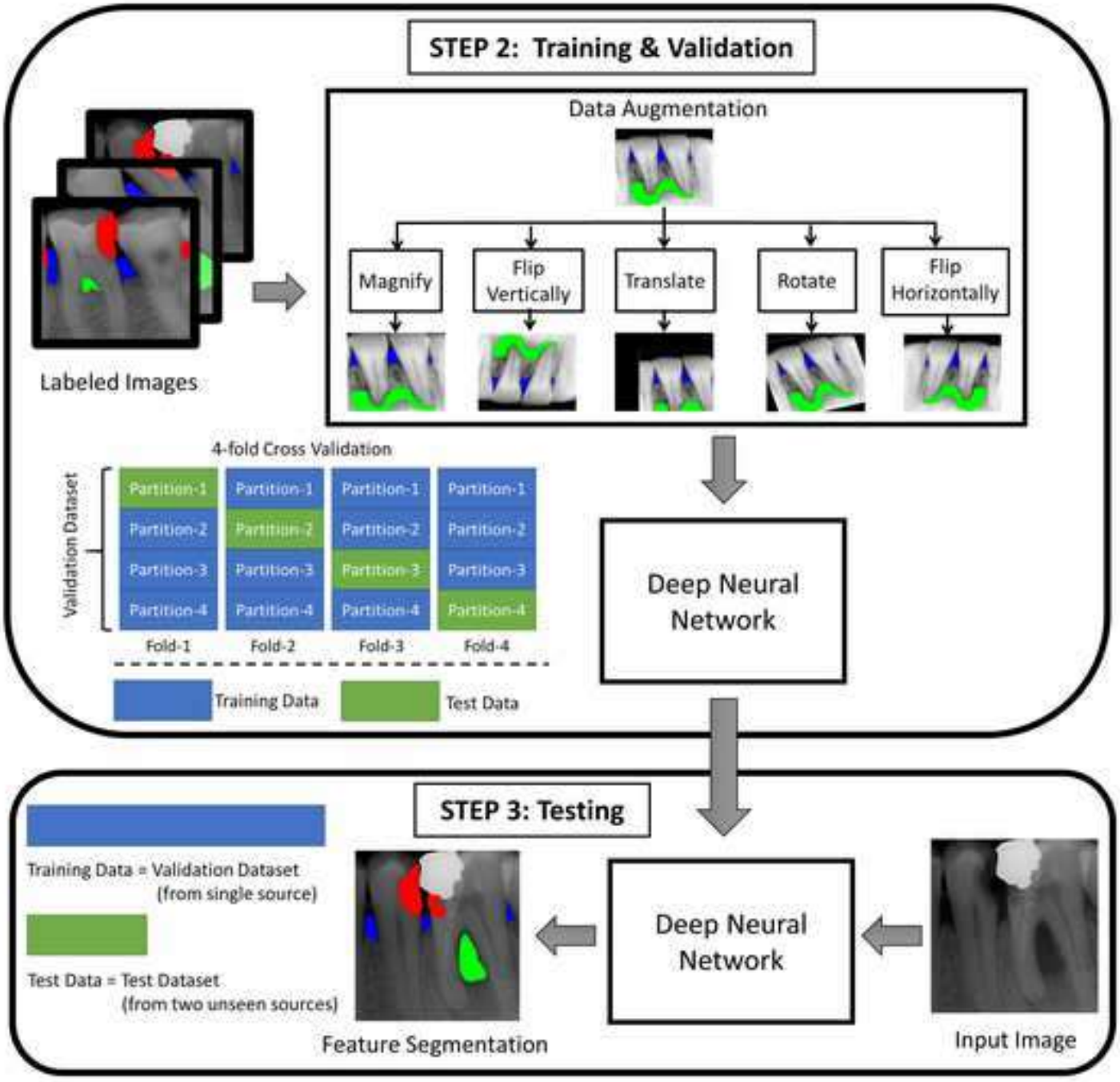


Figure 3

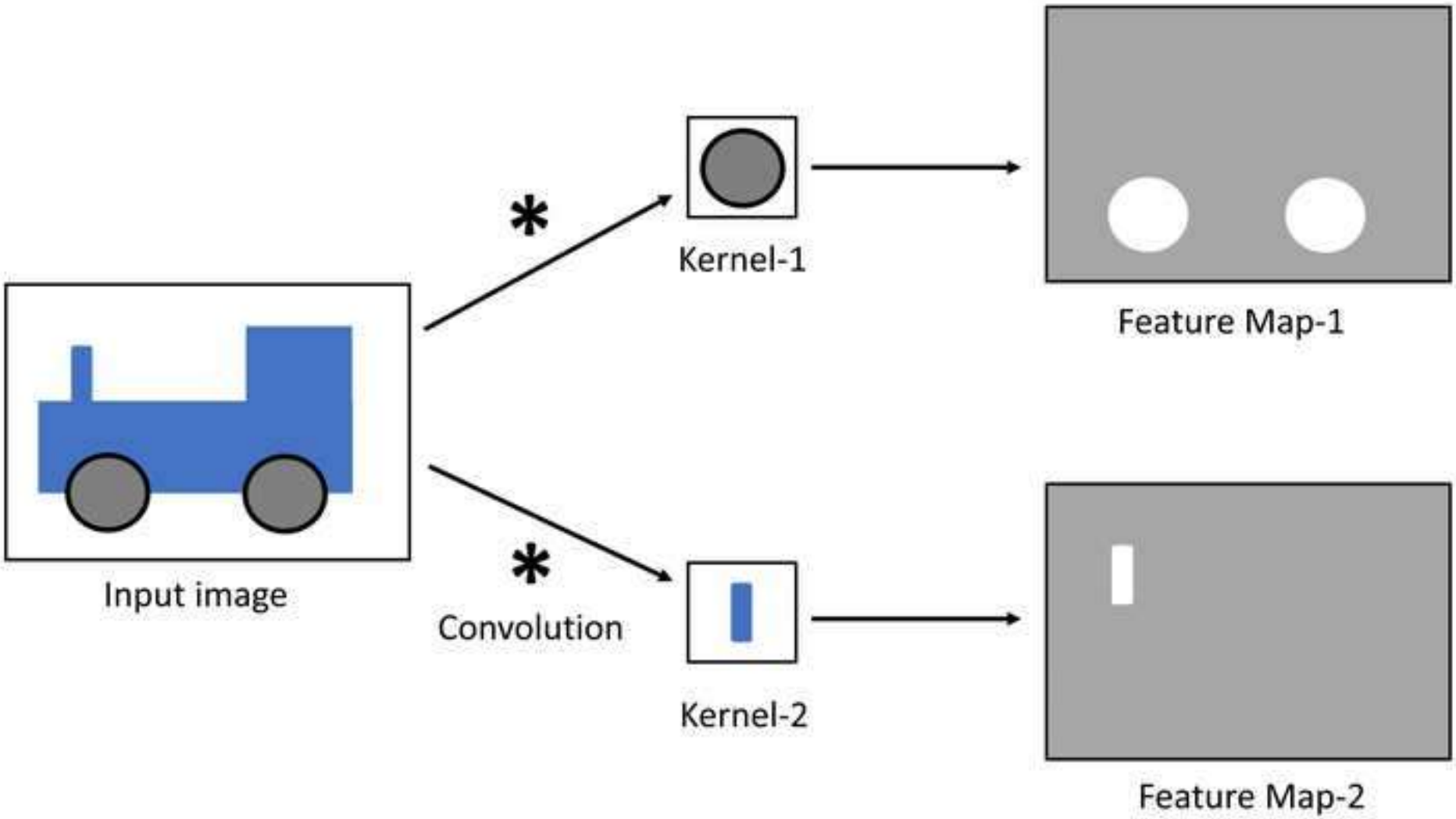


Figure 4

Click here to access/download;Figure;Figure_4_UNet.jpg

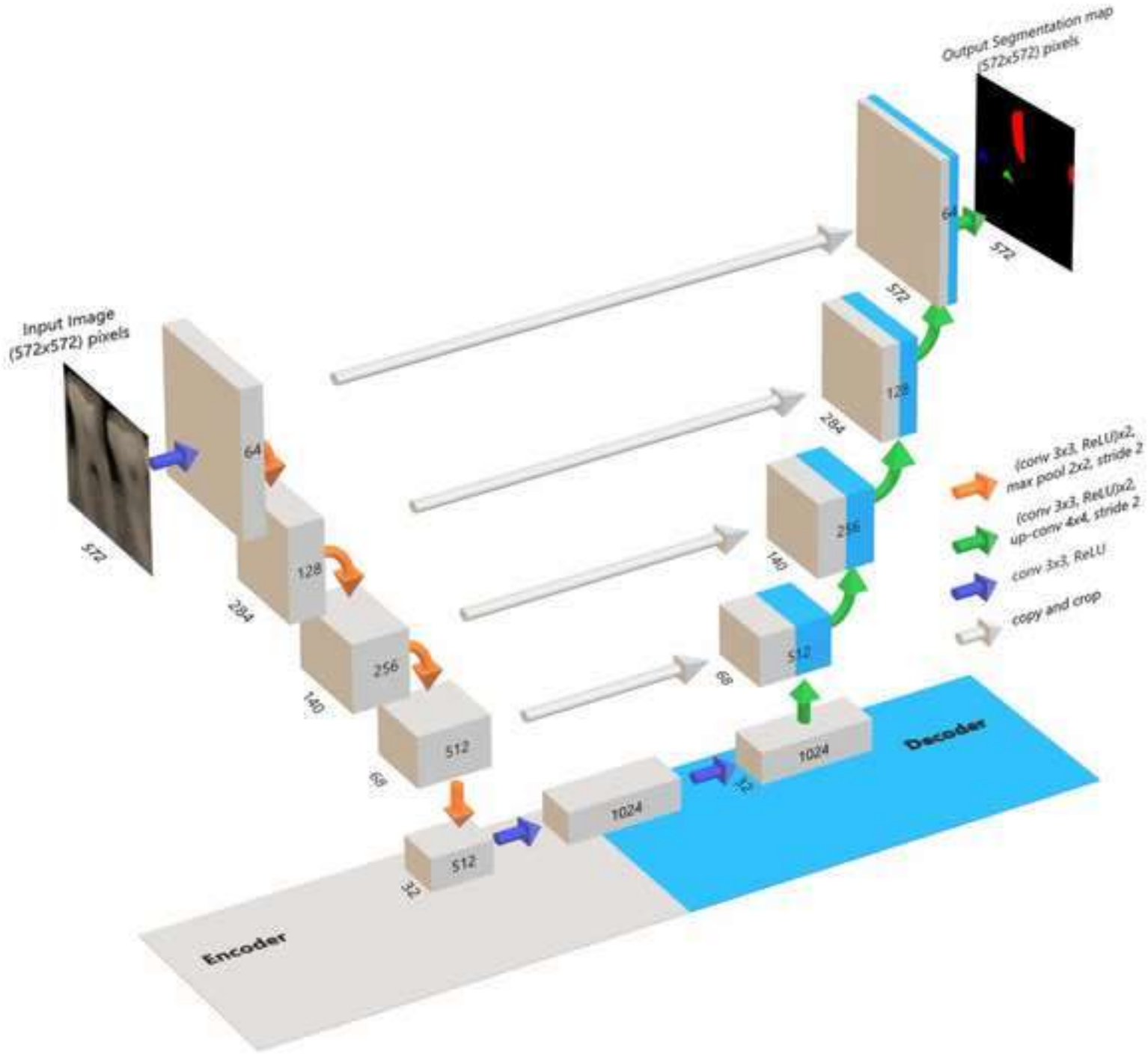
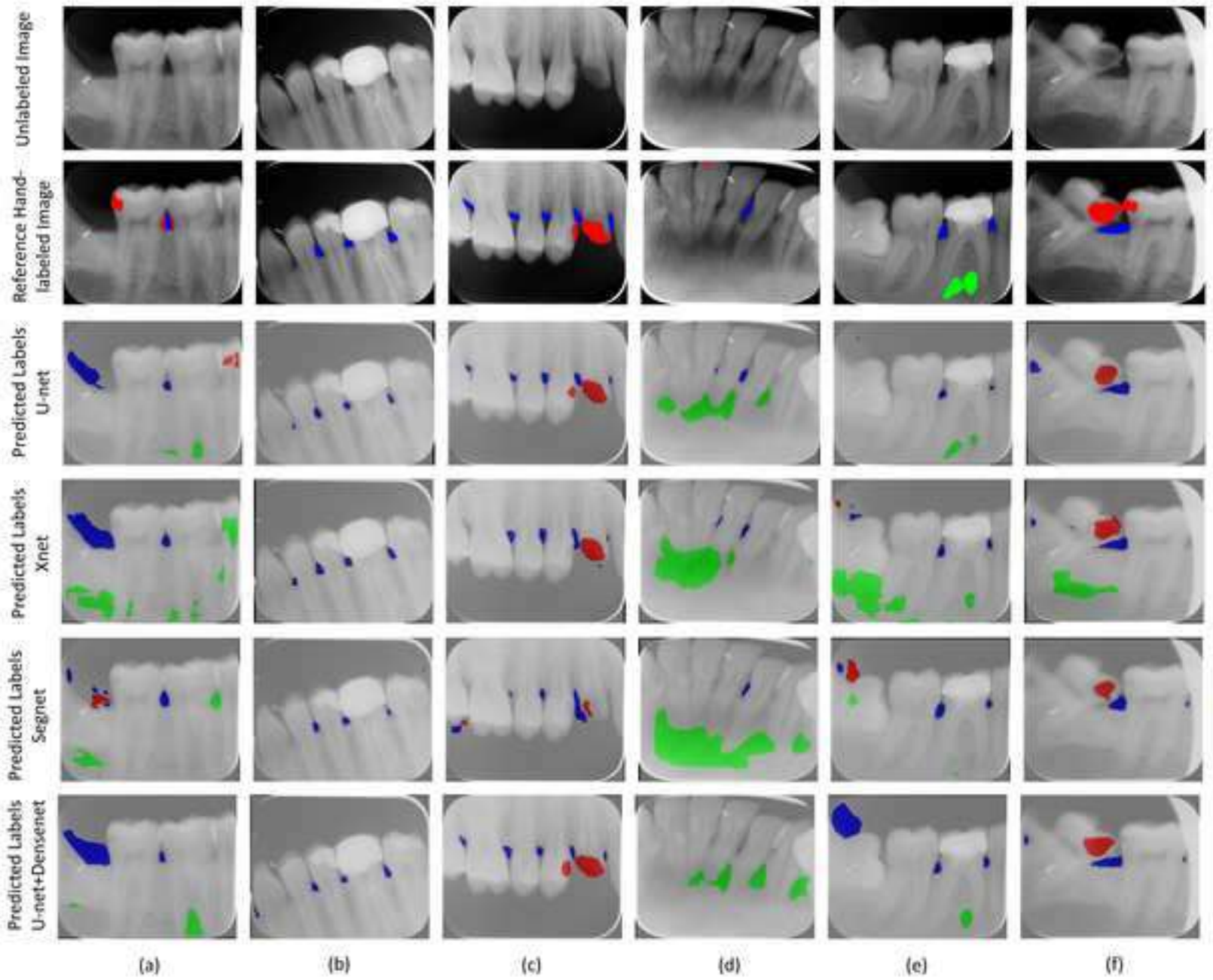


Figure 5

Click here to access/download;Figure;Figure_5_ResultsGrid.jpg



Results (Validation Dataset)		mIoU (averaged over 4-fold cross validation)				Dice Coefficient (averaged over 4-fold cross validation)			
		U-Net	Xnet	SegNet	U-Net + Densenet121	U-Net	Xnet	SegNet	U-Net + Densenet121
Approximate Number of Parameters (in Millions)		8	12	5.5	25	8	12	5.5	25
Types	**Background**	0.940	0.937	0.946	0.962	0.969	0.967	0.972	0.981
	Caries	0.291	0.207	0.119	0.428	0.376	0.279	0.182	0.532
	ABR	0.400	0.386	0.383	0.440	0.493	0.490	0.507	0.556
	IRR	0.235	0.102	0.003	0.173	0.296	0.136	0.005	0.206
Average		0.466	0.408	0.363	0.501	0.534	0.468	0.417	0.569

Table I: Segmentation performance for the validation dataset.

mIoU: mean intersection over union value

Results (Test Dataset)		mIoU				Dice Coefficient			
		U-Net	Xnet	SegNet	U-Net + Densenet121	U-Net	Xnet	SegNet	U-Net + Densenet121
Approximate Number of Parameters (in Million)		8	12	5.5	25	8	12	5.5	25
Types	**Background**	0.981	0.961	0.960	0.972	0.991	0.980	0.980	0.986
	Caries	0.166	0.127	0.050	0.194	0.202	0.169	0.076	0.239
	ABR	0.406	0.338	0.330	0.341	0.540	0.466	0.466	0.472
	IRR	0.055	0.003	0.001	0.027	0.077	0.006	0.004	0.040
Average		0.402	0.357	0.335	0.383	0.453	0.405	0.381	0.434

Table II: Segmentation performance for the test dataset.

mIoU: mean intersection over union value