A Comparison of Deep Learning U-Net Architectures for Semantic Segmentation on Panoramic X-ray Images

by

Rahil Bin Mushfiq
18101552
Rafiatul Zannah
18301027
Mubtasim Bashar
18301046
Md. Nafidul Alam
18101080
MD Aftabur Rahman
18101071

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
School of Data and Sciences
Brac University
January 2023

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at Brac University.

2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.

3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.

4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Rahil

Rahil Bin Mushfiq
18101552

Rafiatul Zannah.

Rafiatul Zannah
18301027

Mubtasim

Mubtasim Bashar
18301046

Md. Nafidul Alam
18101080

Aftab

MD Aftabur Rahman
18101071

Approval

The thesis titled “A Comparison of Deep Learning U-Net Architectures for Semantic Segmentation on Panoramic X-ray Images” submitted by

1. Rahil Bin Mushfiq(18101552)
2. Rafiatul Zannah(18301027)
3. Mubtasim Bashar(18301046)
4. Md. Nafidul Alam(18101080)
5. MD Aftabur Rahman(18101071)

Of Fall, 2022 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on January 19, 2023.

Examining Committee:

Supervisor:
(Member)

Amitabha Chakrabarty, PhD
Associate Professor
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Md. Golam Rabiul Alam, PhD
Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

Digital image processing utilizes deep learning to tackle challenging issues such as image colourization, classification, segmentation, and detection. The medical image analysis field is developing day by day, and segmenting organs, diseases, or abnormalities is a challenging task to complete. Dental disease diagnosis is one of these fields where image segmentation can help gain significant improvements as dentists worldwide face various problems in diagnosing dental diseases with the naked eye. Compared to other medical images, dental radiographic images provide multiple challenges in terms of processing, making segmentation a more complex task. Deep neural network models are used more frequently for various image segmentation applications. U-Net is one such model. Multiple variations and advancements have been created for this network model to serve better performance, mainly on semantic segmentation of medical images. However, comparative studies must determine how well these variants perform in segmenting dental x-ray images. This research uses six U-Net architecture (Vanilla U-net, Dense U-net, Attention U-net, SE U-net, Residual U-net, R2 U-net) variants for segmenting dental radiographic X-rays that are extensively and effectively compared. Some U-Net architectural variations under consideration still need to be evaluated for segmenting dental radiographic X-rays. For all architectures, we used 2 and 3 convolutional layers. We used four types of matrices to compare the models: Accuracy, Dice coefficient, F1 score and IoU. Among the variants, Vanilla-unet with two convolutional layers provided the best Accuracy of 95.56% and IoU score of 88% on the validation set for much lesser time than other architectures. On the other hand, when we use three convolutional layers, dense-unet provides the best Accuracy of 95.94% and IoU score of 89.07% on the validation set. However, most of the examined architectures throughout the dataset showed minor changes when segmentation performance was measured using all four accuracy metrics. This study indicates that U-Net is enough for radiographic X-ray segmentation. Choosing simpler models will save time and money during testing and model creation. Therefore, our suggested approach might aid in making automated dental disease diagnosis models.

Keywords: Dental; Semantic Segmentation; Data Annotation; OPG Image; U-net; U-net Variants; Dice Coefficient; IoU; Architecture Comparison

Acknowledgement

We would like to express our deepest gratitude to the Almighty for guiding and strengthening us throughout the journey of completing this thesis. We are also deeply grateful to our supervisor, Dr. Amitabha Chakrabarty, for his invaluable guidance, support, and encouragement throughout the research process. A special thanks go to Shahriar Hossain, our supervisor's research assistant, for his dedication and contributions to this project. Lastly, we would like to express our heartfelt thanks to our parents for their unwavering love and support, without which this thesis would not have been possible.

Table of Contents

List of Figures

List of Tables

List of Acronyms

This list describes several abbreviations that will be later used within the body of the document

Al	Artificial Intelligence
ANN	Artificial Neural Network
AS	Automatic Tooth Segmentation
BPNN	Back-Propagation Neural Networks
CBCT	Cone-beam Computed Tomography Systems
CNN	Convolutional Neural Network
CRNN	Convolutional Recurrent Neural Network
cSE	Spatial Squeeze and Channel Excitation Block
DCNN	Deep Convolutional Neural Network
DenseNet	Densely Connected Convolutional Network
DGCNN	Deep Graph Convolutional Neural Network
DIFOTI	Direct Image Fiber Optic Trans-illumination
DRU	Dense Residual U-net
DS	Tooth designation and segmentation
ECM	Electrical Caries Monitor
FN	False Negative
FP	False Positive
FPN	Feature Pyramid Network
IoU	Intersection over Union
KNN	K-Nearest Neighbors Algorithm
LS	Landmark-based Tooth Segmentation
MFFR	Multi-scale Feedback Feature Refinement

MRI	Magnetic Resonance Imaging
OPG	Orthopantomogram
$PaxNet$	Panoramic Dental X-ray Network
$QLIF$	Quantitative Light-induced Fluorescence
$R2$	Recurrent Residual
$ReLU$	Rectified Linear Unit
$ResNet$	Residual Neural Network
RNN	Recurrent Neural Network
ROI	Region of Interest
$RRCNN$	Recurrent Residual Convolutional Neural Network
$scSE$	Concurrent Spatial and Channel Squeeze and Channel Excitation
SE	Squeeze and Excitation Network
SGD	Stochastic Gradient Descent
sSE	Channel Squeeze and Spatial Excitation Block
TN	True Negative
TP	True Positive

Chapter 1

Introduction

Dental diseases (i.e., bone loss, fluorosis, decay, gum disease, etc.) have become the most common phenomenon in the field of medical science and therefore are increasing highly in recent times. Health begins with the mouth, according to several medical studies. Consequently, dental health is just one aspect of oral health, including our body's overall health and well-being [13]. However, when we only talk about dental health, teeth are one of the body parts that are overused and have a high resistance to damage and long-lasting architectural durability. Still, it is vulnerable to various illnesses [5]. Dental diseases affect nearly 90% of individuals in the United States, as mentioned in a survey of the National Health and Nutrition Examination [2]. Furthermore, according to specific epidemiological statistics, dental disease is more prevalent in communities with poor socioeconomic positions [3]. Even then, Health and Retirement Studies show that even if an individual's wealth drops by 50% in the US, they pursue dental care for their well-being.

A wide range of dental care is given through hands-on treatment or radiographic images. Since the invention of X-ray imaging, oral radiographic images having three types (Bitewing, Panoramic, and Periodical) have been widely employed [1]. These dental X-rays enable the hidden and inner part of teeth that a dentist's eye can hardly detect. Among the three types of x-ray radiography in dental anatomies, Panoramic x-ray, known as the Orthopantomogram (OPG), is less time-consuming, has the lowest radiation value and causes less patient discomfort. Therefore, using OPG images or panoramic x-ray can be a very impactful approach to detecting dental diseases.

Nowadays, Deep Learning approaches show enormous potential and versatility in every medical field, handling massive datasets using advanced computation processes with reasonable accuracy and proper decision support. Convolutional Neural Network is the most revolutionary algorithms in the field of a lot of sectors. As mentioned in [16], CNN was created by Prof. YannLe-Cun in the 90s era. It was initiated as a machine learning algorithm, a unique form of ANN, developed to perform image analysis and 2D recognition of handwritten digits. Later on, CNN became the most successful application for recognizing and categorizing images.

Direction finding with CNN and CRNN [25], face recognition using CNN [17], classification of different diseases such as Lung cancer, dental disease [19], Speech emotion recognition [26], Handwritten digits recognition [14] etc. can be good examples of CNN in terms of having different layers of computation process and detection, classification and recognition accuracy. There are various forms of CNN out there; one of them is U-net. U-net is one of the breakthrough techniques which overcome the various challenges in the medical imaging field. U-net is commonly used within the medical imaging community. According to [23], In terms of the number of training samples needed, memory requirements, and computing time, u-net, which is trained in an encoder-decoder architecture, has excelled in earlier research. The primary goals of the many U-net variations developed are to increase the segmentation accuracy and effectiveness of feature information passing both within and between layers.

Our research mainly focuses on segmentation. Early semantic image segmentation methods relied heavily on manually created features and their integration with classifiers to categorize pixels at the pixel level. However, the robustness of the constructed feature descriptors has always limited how well these systems function. In recent years, the development of deep convolutional neural networks has hugely promoted the widespread use of medical image segmentation. Medical image segmentation is one of the essential steps in medical image analysis. The segmentation results' shape, size and total area can provide crucial information for understanding early signs of potentially deadly diseases, as described in [28]. For this reason, we selected U-Net, as it has shown massive success in semantic segmentation. Also, a wide variety of variations and advancements have been created. In this study, we have segmented our collected OPG images of teeth using several variants of U-Net to determine the performance gain between them, considering different parameters.

1.1 Research Problem

Traditionally, dentists' visual inspection and non-automated radiographic diagnosis is the approach to diagnosing oral illness. A quicker approach with efficiency is needed in case of treatment for patients to have less discomfort. Unfortunately, visual inspection is a difficult process to identify the oral illness. Moreover, according to [27], both visual evaluation and x-ray imaging assessment are labor-intensive as well as time-consuming in detecting dental decay on the jaw's teeth. Similarly, as mentioned in [20], Dentists look for roughness and discoloration on the tooth surface with visual tactics for caries detection. This detection technique is highly subjective and highly reliant on dentists' expertise.

As stated in [27], above mentioned manual methods should not be considered, given the COVID-19 pandemic situation. Where these approaches are the failure to maintain social distancing which further increases the risk of infection. In the current COVID-19 outbreak, some children with dental caries must adopt social distancing to minimize infection, which results in missing needed treatment for the oral illness.

Misdiagnosis is probable since typical tooth decay detection approaches including ocular inspection and panoramic x-ray tests are dependent on competent practitioners. Medical misdiagnosis has become widespread in recent years, and many people are affected directly or indirectly. Furthermore, because early and concealed caries is difficult to diagnose, the rate of misdiagnosis is significant. [35]. If dental caries is not treated promptly, they can spread over time, causing different oral disorders eventually leading to tooth loss.

In addition, analyzing the x-ray images of teeth is more complex than analyzing other medical images. Because digital radiography employs a tiny amount of radiation, the image quality is poor, resulting in false-negative recording [20]. As, digital radiographs are noisy, finding the edges is difficult. Dental radiography image analysis is made more difficult by visual noise and poor contrast. Furthermore, [8] claims that visual inspection has a lower acuity rate, meaning that man-made assessment alone overlooks a significant proportion of decay. Although image segmentation algorithms have improved over the years, due to variances in the images, they remain complex and demanding operations. Artifacts from the treatment procedure, impacted teeth, varied varieties of teeth, and missing tooth space are all issues. Finding an accurate and proper approach for segmenting dental X-ray pictures remains a difficult task due to these issues.

Dentists continue to face difficulties in accurately diagnosing dental cavities early. Existing caries detection technologies are not universally approved by dentists, and the results are often inaccurate [20]. The main drawbacks of the diagnostic caries monitor include false-positive diagnoses caused by food debris and plaque deposition, tooth staining, and low mineralization, which leads to an incorrect diagnosis. Because the Electrical Caries Monitor (ECM) produces a significant number of false positives for stained teeth, it should be used with caution. Direct Image Fiber Optic Trans-illumination (DIFOTI) and QLF (Quantitative Light-induced Fluorescence) become unsuccessful, because of having the interpretation of images by dentists with costly equipment and being a complicated method with manual examination respectively made hidden carries impossible to detect.

Decision-oriented algorithms show a more promising result and accuracy than the traditional methods of teeth segmentation, as per [20]. The image contains both teeth and bones that display similarly and using an engineering tool is an easier approach to detecting the tooth. As a result, computer-assisted methods for analyzing detection and treatment may find the images more convenient.

Taking panoramic dental x-ray as the base of this system we can solve half of the problems. As the panoramic x-ray emits less radiation as well as taking this type of x-ray would provide patients deserving comfort. Moreover, the panoramic x-rays will be preprocessed with the help of some algorithms. After that, with the help of a decision-making algorithm which is a Convolutional Neural Network in this case would be convenient for having a good detection accuracy building up a system that would predict dental diseases accurately.

1.2 Research Objective

We aim to perform segmentation on panoramic x-ray images, extracting tooth-eliminating jaw regions from the images. The goal is to gain a practical outcome for our domain by comparing u-net variants. This comparison will show us if a simple Vanilla-unet architecture would give us better accuracy in less time or if more complex architecture would give us better accuracy in much more time. In our proposed model, we use six variants of the u-net architecture. We will see which model will perform the segmentation of teeth more efficiently. The objectives of our proposed research are:

1. To increase the completeness of electronic dental data and save time.
2. Implement six models based on u-net architecture and apply them for segmentation tasks.
3. To obtain better accuracy.
4. To evaluate the models.
5. To compare the outcome of these models based on training time and accuracy matrices.

1.3 Research Motivation

Dentists can identify cysts, carries, and other issues with the help of X-rays that require more information than can be obtained through a patient's direct examination. Dentists can assess the overall dental structure from the X-ray images and design patients' treatments. Nevertheless, analyzing an X-ray is not a simple task. However, analyzing x-ray images by eyes is no easy task. A professional must have years of training and experience before providing an accurate diagnosis. An automated system can help to analyze X-rays more accurately. Most research articles on automatically examining dental X-rays relied on custom feature extractors. These works mostly ignored Panoramic X-rays, perhaps because they provide unique challenges. That is why we decided to seize the chance to offer our unique perspectives on this study area. Image segmentation is a tremendously difficult task in the medical field due to the many imperfections in the images. In terms of medical image analysis, deep learning approaches are the most promising. A deep learning methodology can automatically learn the rules from a dataset and solve the primary problems in that domain with different models. The paper [29] serves as a significant motivation for this study. It is centred on several models built on U-net architecture and provides extensive descriptions of how they are constructed and operated. Also, it shows the excellencies of these models in performing image segmentation tasks. So, In this situation, u-net can reveal structural and functional information about the tooth, aiding in the analysis of x-ray images. Moreover, our work might help in the case of detection in the future.

1.4 Research Outline

The thesis is organized in the following manner: We gave an overview of the potential of deep learning models in panoramic x-ray image segmentation in Chapter 1. We've also discussed our research objectives and our motivation to proceed with our work. Chapter 2 is the literature review focusing on relevant work and existing methods based on our topic. In Chapter 3, we showed how we made our dataset and its description and how to preprocess it to suit the models. In Chapter 4, we showed our workflow. Moreover, we discussed all the architectures that we employed in our thesis. Then, we demonstrated how we implemented those architectures and described the evaluation methods. We examined and then analyzed the data in Chapter 5. Finally, Chapter 6 concludes our whole work. Also, we talked about our future intentions.

Chapter 2

Literature Review

The operation of our brain is imitated by the mathematical models of Neural Networks. Its computing system is composed of various basic, highly interconnected processing components that process information by responding to external inputs with dynamic responses. Artificial neural networks are built based on a similar concept. Although neural networks offer several characteristics that make them suited for a variety of tasks in healthcare, such as performing segmentation on various types of medical images, research on this application in dental care, particularly in the dental image, is relatively minimal.

2.1 Related Works

Dental illnesses have a high chance of spreading over the world, especially among adults. In comparison to other medical images, analyzing dental X-ray images presents some challenges, making segmentation and detection more complicated. For a caries detection system, both segmentation of teeth and the disease detection method improve the accuracy and reliability.

As mentioned in [12] that, it resolves the challenge of initializing the tooth model by itself, and the findings demonstrate that the tooth morphologies may be extremely closely matched, particularly if the set of teeth is accurately specified. The teeth segmentation problem is solved in two phases using RFRV-CLMs, which is one of the most recent contributions in the statistical model field. The first stage is estimating a few teeth and related mandibular areas that are being used to begin the search for individual teeth, and phase two involves searching each tooth individually. To identify missing teeth, an automated quality-of-fit measure was devised. While detecting the contour of existing teeth, a median point-to-curve error of $0 : 2$ mm for every single tooth is displayed by the system.

The paper [15] focuses on convolutional neural network, which is based on the U-net model and is used to create a model for teeth segmentation from panoramic images. The model has quite a commendable competency in the case of segmentation. They made the following alterations to the U-Net architecture: they applied batch normalization prior to every max pooling, up-sampling and concatenation layer instead of dropout during training. Moreover, they utilized the Nadam optimizer for the optimization. For research purposes, 1500 panoramic radiographs and

7 were chosen. They achieved a dice score of 0.936. By taking into account the benefits of both residual networks (ResNet) and DenseNet, they suggest an effective network architecture in this study [23]. While using much fewer model parameters than DenseNet, our approach adds more skip connections than ResNet. They Use two datasets to test the suggested approach. For the ISIC 2018 dataset and the brain MRI dataset, they gained a mean dice coefficient of 0.861 and 0.8643, respectively. Again, in [28], they provide a brand-new network for segmenting medical images called the MFFRU-Net. They create an easy-to-use multi-scale feedback mechanism. Through upsampling and 1×1 convolution, the feature maps from the decoder are given back to the encoder, combining several high-level and low-level features to create more relevant features. They used a public image dataset to assess their proposed MFFRU-Net. MFFRU-Net has a 96.78% accuracy rate and a 98.56% AUC, respectively.

Research done in [30] is centered around the segmentation of the 3D image. For end-to-end learning of tooth instance segmentation in 3D point of ios cloud data, a new deep learning-oriented computational model named Mask-MCNet is introduced in this research. The suggested model separates the points that are relevant to each distinct tooth instance while also predicting each tooth's 3D bounding box to localize each tooth instance. This property results in highly precise segmentation that is necessary for clinical practice by preserving the intricate context of data, such as the little curves in the boundary between adjacent teeth. They made use of two datasets that were gathered from two distinct kinds of scanners. The first dataset includes 120 optical images of odontiasis from 60 adult patients, including lower and upper jaw images, which were used both for training and testing. The second dataset consists of 48 optical images of 24 adult people and is exclusively used to assess the robustness of MCNet's to various scanner types. The outcomes demonstrate that the Mask-MCNet beats modern models by reaching a tooth instance segmentation score of 98% IoU, which is extremely similar to the performance of a human expert. Similarly, the paper [32] proposes a hierarchical multi-step model based on deep learning, which automatically identifies and segments 3D individual teeth from dental CBCT images. To get over the computational difficulty posed by high dimensional data, it generates panoramic photos of the upper and lower jaw images on its own. Following that, 3D individual teeth's loose- and tight- ROIs are captured from the acquired 2D images. They used 97 dental 3D CBCT images to do the research. They got a 93.35 F1 score and a 94.79 Dice coefficient percentage for the study.

The study in [33] looks towards lightweight deep learning techniques for segmenting dental X-ray images. This research proposes a novel lightweight knowledge distillation neural network technique. Knowledge distillation is typically a method of transferring information from heavy-duty teacher models to lightweight student models that imitate teacher models. They propose an attempt to retrieve reliable data from a teacher network using a knowledge network. They referred to it as a knowledge consistency neural network for simplicity (KCNet). In total, 1321 dental panoramic images were employed in this research. As their student and teacher networks, respectively, they selected U-Net and ESPNet-v2. There are 432 training images, 111 validation images, and 778 testing images. In terms of the IoU score of 80.4% and the Dice coefficient of 89%, it delivered the best performance. Similarly, the study done in [31] assesses the precision and effectiveness of deep learning-based automatic teeth segmentation in digital dental models. A DGCNN-based algorithm was used to do this research. Three different methods were used to compare electronic dental models: (1) AS (automatic tooth segmentation), (2) LS (landmark-based tooth segmentation) and finally, (3) DS (tooth designation segmentation). Five hundred sixteen dental models were used to train a deep learning system to segment teeth, and 30 dental domains were used to evaluate the precision and efficiency of the segmentation. The accuracy of tooth segmentation was 97.26%, 97.14%, and 87.86% for the AS, LS and DS, respectively.

Furthermore, the study [34] shows the viability of the SWin-U-net CNN model for segmenting teeth on panoramic x-rays. SWin-Unet is an encoder-decoder system that uses transformers and is shaped like a U with skip connections. In SWin-Unet, a symmetric encoder-decoder structure is built using jump connections. It uses a local to a global strategy for self-attention. Moreover, it builds a patch-expanding layer to increase sampling and feature dimension without using convolution or interpolation techniques. For research purposes, 100 panoramic radiographs of adult patients were randomly chosen. They used 10 panoramic radiographs for testing and 90 for training. They achieved an accuracy of 88.52% using SWin-Unet.

The paper [21] states that using a Genetic Algorithm for automated teeth extraction and classification from panoramic radiographs. The system performs image enhancement preprocessing in two steps - (i) Preprocessing for Initial ROI Detection and (ii) Preprocessing for Last ROI Detection. Then at the end of the extraction process, it initiates Jaw separation using the middle point method. Finally, after drawing 30 random lines, it applies the Genetic algorithm to determine the Best Fit line, completing the Extraction process with a 77.56% accuracy (maxillary 81.44%, mandibular 73.67%). A similar approach has been demonstrated in [22] where a genetic algorithm extracts teeth from panoramic images. They used the Capsule Network classifier for dental caries diagnosis and PaxNet for dental disease identification. The raw images are considered and go through preprocessing, ROT extraction, and jaw separation to be prepared for the Genetic algorithm to separate teeth with vertical lines. After that, these extracted teeth are fed into PaxNet (Panoramic dental x-ray network) for caries detection, which has a feature extractor (Encoder, CheXNet, InceptionNet) and CNN classifier modules. The architecture of PaxNet contains four layers in feature extraction and 2 layers in Capsule Network. Initially, teeth extraction is done by running some algorithm on 42 panoramic images where jaw extraction accuracy is 95.23%. A genetic algorithm is then applied to extracted jaws to isolate the tooth, and the accuracy of extracting maxillary and mandibular teeth is 81.44% and 73.67%, respectively. PaxNet is trained with healthy and unhealthy samples. Finally, using a dataset of 5948 extracted tooth images, the training and testing accuracy of Pax Net is 91.23% and 86.05%, respectively, while having an F0.5-score of 0.78.

Mask R-CNN could be used to assist dentists in an instance, segmentation of teeth and diagnosing problems. According to [9], a faster version of R-CNN conducts instance segmentation of teeth. First, features from ResNet101 are extracted, and these features are combined to form an FPN that defines anchors and extracts ROIs.

These removed ROIs are then molded into the same size, referred to as 9 the tooth. The training portion of the dataset was completed in two parts-Adam Optimizer and SGD. The segmentation requires quick weight adjustment with the values of 103, 1 as 0.9, 2 as 0.999, and 108, which the Adam optimizer can provide. The SGD is used to fine-tune the weights without any momentum, with $10 - 6$ as the learning rate. The MSCOCO dataset is utilized, which contains 193 buccal panoramic x-ray images divided into 10 categories. After training with these images, the Mask RCNN achieved 98 percent accuracy and a 0.88 f1-score. Similar utilization of this algorithm has been mentioned in the image segmentation phase in [24], where they used it from the sample library. The Al model successfully reached 90% of diagnosis accuracy. The paper demonstrates making up an intelligent dental Health-IoT system that is organized and has 3 layers of services. CMOS-1 megapixel sensor is used, allowing the system to have images that are being processed or enhanced, and then the teeth segmentation from the image is performed. After making the training data set using the semi-automatic labeling method, the clinical images were labeled by the detector, classifying them into 7 types of dental diseases. The detector's function includes visual enhancement, coarse localization, and classification, and following these functionalities with labeling errors and classification errors, the system performs Artificial Screening.

This article [8] was addressed by establishing a unique segmentation strategy of the image that both solved the shortcomings of current approaches and produced more significant outcomes. The suggested system comprises three primary stages; pre-processing, segmentation, and analysis. These stages suggested technique for the segmentation phase is based on a two-phased enhanced level set (LS) method. In the latter stage, various Back-Propagation Neural Networks (BPNN) algorithms are applied to utilize the algorithm known as ”Traingda” under diverse setups. In addition, a new strategy for isolating individual images of the segmented teeth has also been presented using an integral projection method and a region-based feature map constructed on every tooth to retrieve the local data and, as an outcome, identify the area which contains caries. When tested on the 120 oral radiographic X-ray pictures, the proposed segmentation algorithm obtained an all-inclusive score of 90.83% and a remarkable outcome of 98% accuracy in detecting the total 155 segmented teeth.

Similarly, as per [35], CariesNet is a new deep learning architecture created as a U-shape network with an extra full-scale axial attention mechanism. It is used for segmenting caries types from dental Radiographics. From 1159 x-rays, three types of labeling are applied to 3217 caries locations. The feature extraction process from multi-level CNN is combined with the U-shaped framework. They rely on Res2Net as a reliable backbone. Experiments reveal that their technique can segment three degrees of caries with a mean Dice coefficient of 93.64% and a 93.61% accuracy.

Ref	Task	Classifier	Dataset	Accuracy
[8]	Computer-aided Caries detection	BPNN	Private: Total 120, 84 training-set, 36 testing-set. From university students of age 25 to 35.	90.83% for segmentation algorithm, 98% for detection phase
[9]	Instance segmentation of teeth	Mask R-CNN	Public: MSCOCO dataset of 193 buccal panoramic x-ray images categorized in 10 categories	Accuracy is 98%, F1-score is 88%, Precision is 94%, Recall is 84%, Specificity is 99%
[12]	Adult OPG Image Teeth Segmentation	RFRV-CLMs, quality-of-fit measure	Private: Total 346, training 261, testing 85	Each tooth has a median point-to-curve error of 0 : 2mm.
[15]	Accurate Segmentaion of Dental Panoramic Radiographs with U-nets	Fully Convolutional neural network based on U-net architecture	1500 dental panoramic radiographs	a Dice score of 0.936
[21]	Teeth extraction on Panoramic images	Genetic Algorithm	Public: a dataset of 42 images	Accuracy is 77.56%
[22]	Dental caries detection	Genetic algorithm for teeth extraction, PaXNet	Private 470 Panoramic images	By Genetic algorithm, maxillary and mandibular teeth extraction accuracies are 81.44% and 73.67%. PaXNet accuracy is 86.05% and the f0.5 score is 0.78
[23]	DRU-NET: An Efficient Deep Convolutional Natural Neural Network for Medical Image Segmentation	DRU-NET, Dense and Residual	Public dataset. 1) the ISIC 2018 Grand Challenge: 2594 RGB images of skin lesions, 2) a local brain MRI dataset: 310 2D slices	Mean dice coefficient of 0.861 for the ISIC 2018 dataset and 0.8643 for the brain MRI dataset.
[24]	A smart IoT Platform for home-based dental healthcare services	CNN (Mask RNN)	Private: 12600 clinical x-rays from 10 clinics	Accuracy is 90%

Ref	Task	Classifier	Dataset	Accuracy
[28]	Multi-scale Feedback Feature Refinement U-net for Medical Image Segmentation	The Multi-scale Feedback Feature Refinement U-Net (MFFRU-Net)	Public dataset DRIVE, LUNA and Montgomery County	96.78% accuracy rate and a 98.56% AUC
[30]	Mask-MCNet: Tooth instance segmentation in 3D point clouds of intra-oral scans	Mask-MCNet	2 datasets: 1) 120 optical scans 2) 48 optical scans	IoU 98%
[31]	Accuracy and efficiency of automatic tooth segmentation in digital dental models using deep learning	Dynamic graph convolutional neural network-based algorithm	516 dental models for training and utilized 30 dental models for testing	97.26%, 97.14%, and 87.86% for the AS, LS and DS respectively.
[32]	A fully automated method for 3D individual tooth identification and segmentation in dental CBCT	A deep learning-based hierarchical multi-step model	97 dental 3D CBCT images	F1-score of 93.35% a Dice coefficient of 94.79%
[33]	Lightweight deep learning methods for panoramic dental X-ray image segmentation	KCNet	432 training images, 111 validation images, and 778 testing images	IoU 80.4% and Dice 89%.
[34]	Transformer-Based Deep Learning Network for Tooth Segmentation on Panoramic Radiographs	SWin-Unet deep convolutional neural network	Private: PLAGH-BH 100 panoramic radiographs	88.52%
[35]	Segmentation of multi-stage caries lesion	U-Net (convolutional neural network)	Private: 1159 datasets	A mean Dice coefficient of 93.64 percent and A 93.61 percent accuracy.

Table 2.1: Summary of all papers related to our research

Overall, the studies discussed above mainly focus on medical image segmentation, more specifically, the segmentation of dental images. Although some of them focus on other types of medical images [23], [28] and some provide both teeth segmentation as well as disease classification [9], [24]. The studies done in [31], [32] are based on 3D image segmentation, which is not our area of concern at this moment as we are focusing on 2D panoramic x-rays. Some of the research introduced new novel models for segmentation [15], [31], [32] by combining features from U-net variants or by introducing new features.

In this study, we want to compare six U-net variations on dental panoramic x-ray images to evaluate their segmentation performance based on the architecture complexity and total model training time. Though some of the studies above used modified U-net structure and a combination of U-net variants for segmentation, our study aims to compare some U-net architecture variants to figure out which network model is best for our domain for segmentation operation. We want to evaluate the impact of the number of convolutional layers by analyzing two and three levels per block. This study aims to develop observations about semantic dental panoramic image segmentation via U-Net architectures to be used on any dental dataset and save a lot of time for later research.

Chapter 3

Dataset

In this section, we have talked about the dataset that we have used for our research purpose. We want to elaborate on the fieldwork collection process, the used device and process of getting the pictures and then how we processed it into a comprehensible format to feed it into different deep learning models for finding the best models for our purpose.

Deep learning is mainly applied to unstructured data like images to extract information from it. So, we have to find the OPG (Orthopantomogram) images, mostly called the panoramic x-ray images, to have a proper dataset for our models.

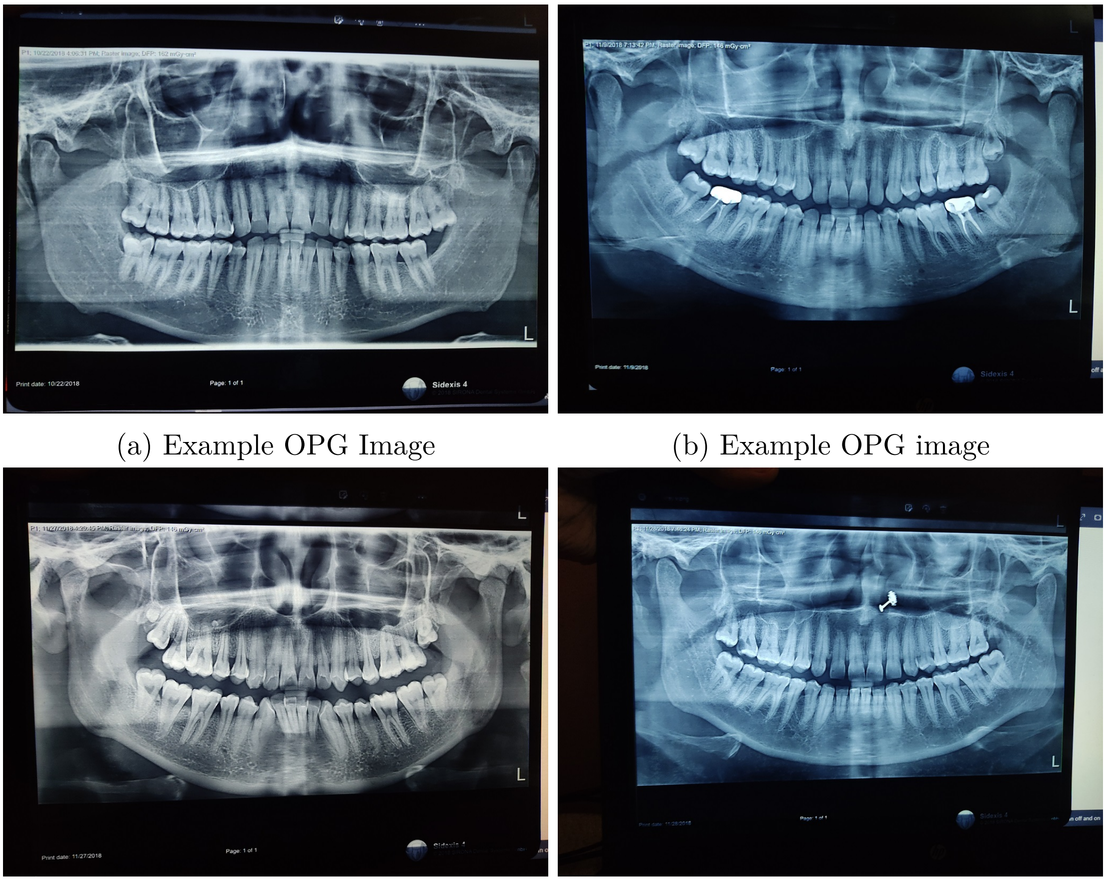
3.1 Data Collection

Our work purpose was to develop a complete dataset of our own. So, after being consulted in many dental clinics, we convinced one of the renowned dentists of Bogura, Bangladesh, Dr. Ashique Mahmud Igbal (BDS, Dhaka Dental College) who allowed us to gather the OPG images of the live patients as well as some pre-stored OPGs of his dental Clinic, IQBAL'S Dental Clinic, with the condition of not sharing the personal information of his patient's name, age and address.

Finally, after getting permission to get the OPG images of the patients, we started fieldwork capturing the OPG images of the live patients with corresponding information. We used our mobile phone cameras. The device used is a Xiaomi Redmi Note 9 Pro with a 64 MP camera for capturing the raw images of the patients. In the first 3 months, we captured 250 images of different patients and manually selected 200 images eliminating blurred images. In the last 2 months, we captured 250 images and manually selected another 189 images. Finally, after merging the images, we got 389 clinical images of patients of the best quality. In the image collection, we found the versatility of patients from children of $5-6$ years to older, both male and female patients of $60-70$ years. Nevertheless, we mostly got images of middle-aged patients.

3.2 Data Sample

As data, we have used the Panoramic X-ray images or OPG images. Some of the samples are given below:



(a) Example OPG Image

(b) Example OPG image

(c) Example OPG image

(d) Example OPG image

Figure 3.1: Data sample

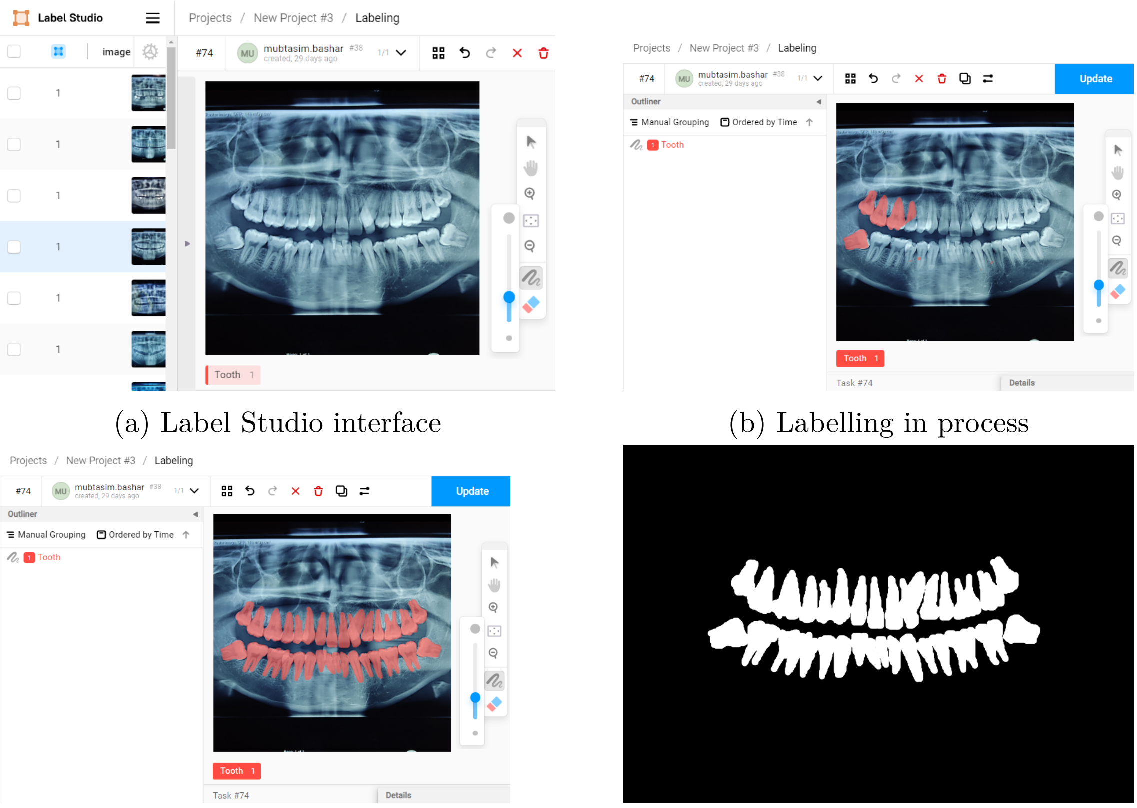
3.3 Data Description

In the two stages of data collection, we get a total of 389 in our dataset, where all the images are from different individuals. From visual inspection, we can see that some images have “blue” tints, and some have “grey” tints. These images are of various aged people, including children, men, and women. Our dataset shows that some x-ray contains 32 teeth; some have less than that, which differs from age to age, man to woman, or even child. Children aged 6 to 10 tend to have different patterned and missing teeth. Overall, our dataset contains a good variety of data.

3.4 Data Annotation

Label Studio is used to label the 389 panoramic x-ray images. We used the in-built Computer vision template of the label studio named ”Semantic Segmentation Using

Mask" for the data labelling of the OPG images. At the very beginning, we selected our very own colour format (R=255 G=76 B=66) for our labelling. We set up some minor changes in the Labeling Interface for the convenience of our work. After the data was imported into the label studio, we set up the region named the region 'Tooth'.Then for the labelling, we eventually masked up the regions or areas that are precisely the teeth areas of a particular x-ray image. We masked the images one by one and generated each image's 'Ground Truth' or mask. Finally, we got precisely 389 masks for each x-ray image after annotation.



(a) Label Studio interface

(b) Labelling in process

(c) Labelling done

(d) Generated mask

Figure 3.2: Data annotation

In figure 3.2a, we have shown the Label studio interface and a particular image that has not been annotated. Then, in the next stop, in figure 3.2b, we have shown the process of labelling the teeth one by one using the brush tool. After that, in 3.2c, we showed the full image after labelling, which finally generates a mask or Ground Truth in 3.2d after exporting from the label studio.

3.5 Data Pre-processing

Focusing on the tooth part of each x-ray and highlighting the valuable details, a few pre-processing steps are considered in the figure 3.3. After creating a dataset from scratch, some steps are implemented before extracting teeth as the primary region within images. The collected raw x-ray images have dimensions of 4000 × 3000 pixels. As we are using different variants of the Unet network model for the Semantic segmentation process, the 4000 × 3000 pixels dimension is unsuitable for the Unet architecture to take as input. So, we have gone through some steps to make the images suitable for our constructed 256 × 256 U-Net models.

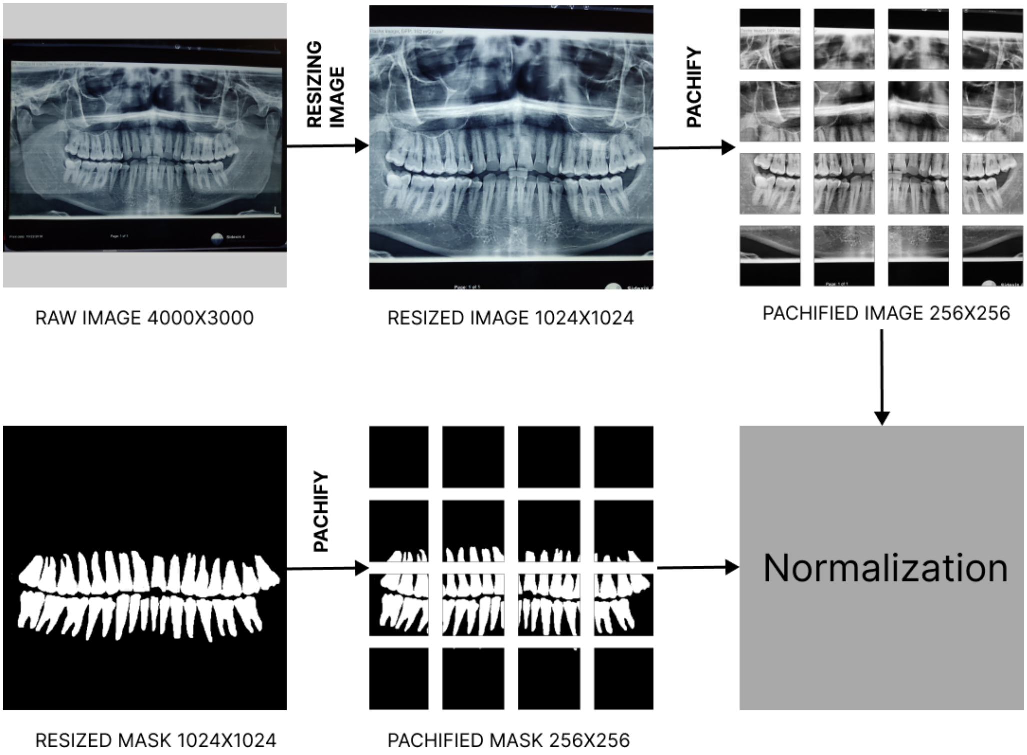


Figure 3.3: Pre-process steps

3.5.1 Crop and Resize Image

After taking the raw OPG images from the Dataset, we used manual cropping to make squared-sized shapes. At the beginning of the pre-processing step, we crop the images to eliminate irrelevant backgrounds and resize images into 1024 × 1024 pixel dimensions for the convenience of unet models.

3.5.2 Patchify Data

U-net architecture with patch-based data is more convenient in the case of images with large dimensions, giving better accuracy. We need to convert the raw (1024 × 1024) images into patched images with 256 × 256 pixels for the proposed models to work. Using Patchify, we divided each image into 4×4 sub-parts of 256×256 pixels each and constructed patched images. Finally, after the Patchify, we get 389 × 16 = 6224 patched images and masks with 256 × 256 pixels, which will be used for training and validation.

3.5.3 Normalization of Data

Our proposed method of normalization of the data is Linear Scaling (equation 3.1). As the images are converted into an array, their values are $0 - 255$ for each pixel. So, using an algorithm like MinMaxScalar, KNN-normalization and so on is unnecessary.

So, we will divide the data by 255 to normalize our data, considering 0 as the minimum value, and 255 as the maximum value.

$$x^{'} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{3.1}$$

3.5.4 Data Classification

With a random state of 42, we divided the dataset into an 8 : 1 : 1 ratio for the train, test and validation set where we have kept the data training size = 0.80 and test data = 0.10 and validation data = 0.10.

Training Set

A training set is a collection of data for any deep-learning model. It consists of input-output pairs, where the input is patched images and the output the patched masks or ground truth. We trained our models with 80% patched images and masked data in pairs so that these models can make the prediction closer to the actual value in performing segmentation. The more training data we provide for our models to learn, the more accurate predictions they can make when used with a validation set to evaluate all unet variants' performance. For our case, we used 4978 patched images and corresponding masks from the overall dataset as a training dataset for training. The training data set is 80% of the total data.

Testing Set

A testing set is a collection of input-output pairs used to test a deep learning model's performance and to estimate how well the model will generalize to new, unseen data. We have used 623 patched images and masks, 10% of the overall data that the unet architecture variants have not seen before, to evaluate the model performance on the unseen data.

Validation Set

When developing a model, a validation set is a set of data separated from the training set and used to evaluate how well the model performs. The validation set is used to tune the model's hyperparameters, which are the parameters set before training the model and cannot be learned from the data. The model's performance on the validation set helps to determine the optimal values for these hyperparameters. We have used around 623 patched images and corresponding masks from the overall dataset, 10% of the overall data, to tune the hyperparameters to train our unet variants with optimal values.

Chapter 4

Research Methodology

In this chapter, We will go through our model methodology, the Unet architecture variants we utilized, and the testing, validation, and evaluation techniques employed in this section's study on panoramic X-ray images in detail.

4.1 Working Progress

In figure 4.1, the flowchart for our thesis work is shown, which includes data collection, data preprocessing, data splitting, 6 different variants of U-net architectures implementation, testing & validation, evaluation of performance, and comparison of the architectures and analysis.

This research starts with collecting data, so we did fieldwork to convince different dentists to share their OPG images for our study. Then, we got a response from Dr. Ashique Mahmud Igbal, a dentist from Bogra who was interested in our research and helped us by sharing the OPG images with the condition of not revealing his patients' information. Then these x-ray images are taken with a smartphone. As the pictures were manually taken by hand, some unwanted background was also clicked with the image. So we cleaned 389 data and resized it to 1024 × 1024 pixels. We later patched each picture to 256 × 256 pixels with a non-overlapping approach to avoid losing any pixels throughout the model training. Furthermore, we used this patched data and split them into train test and validation sets with an 8 : 1 : 1 ratio.

For training, we used six different Unet variants (Vanilla U-net, Attention U-net, Dense U-net, R2 U-net, Residual U-net, SE U-net), and to check performance, we used a validation set. We then used four accuracy matrices (dice coefficient, IoU, f1 score, and accuracy) on the validation set. Then we used a test set to evaluate the model's performance on unseen data.

Moreover, we compared each variant's accuracy over time to find an optimal solution for our domain. Lastly, we took some significant images, patched them again, and predicted on that image, and after segmentation, we unpatched it to see segmentation on a large image.

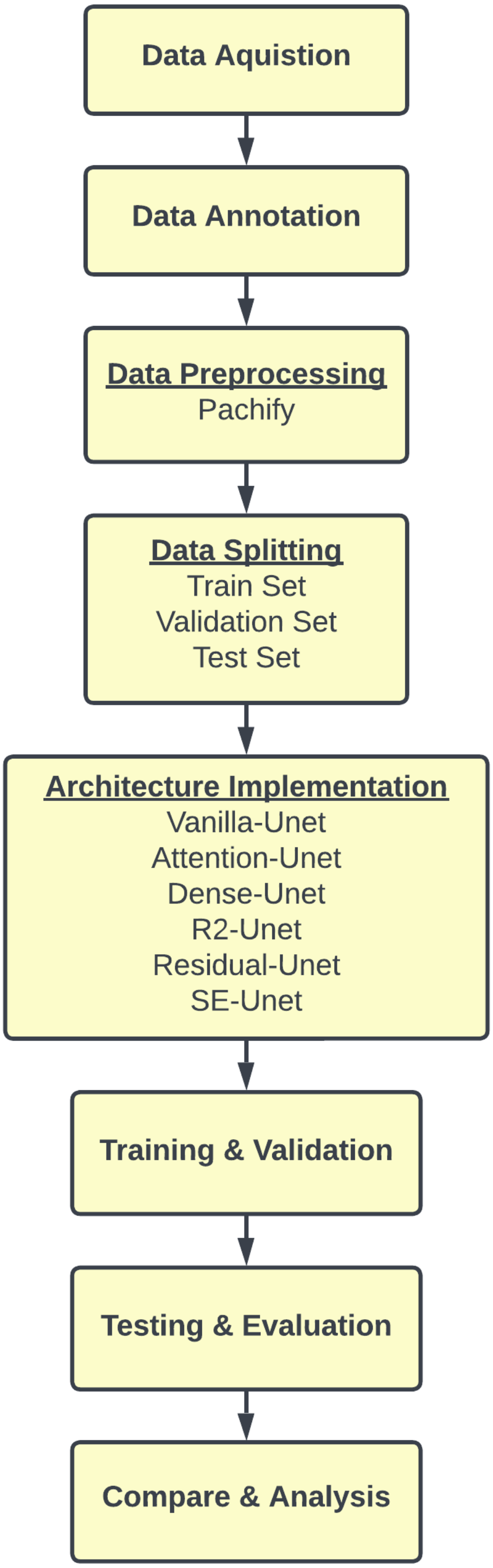


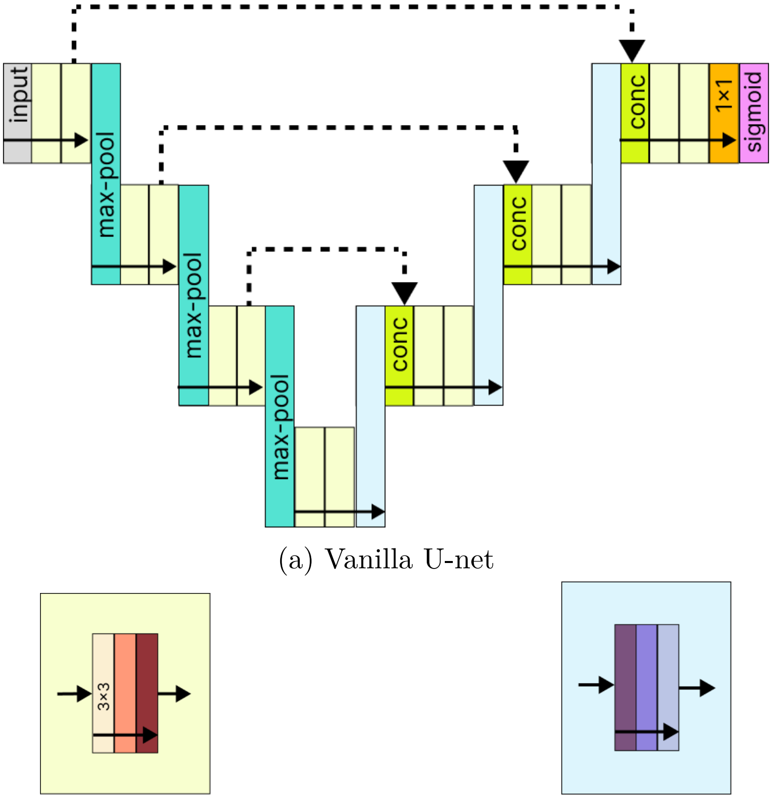
Figure 4.1: Work plan

4.2 Used Architectures

Unet was introduced in 2015 by Olaf Ronneberger [4] to perform segmentation on biomedical images. According to [29], U-net is primarily used in different domains of biomedical images, such as CT scans, MRIs, microscopy, and X-rays. This architecture is mainly used for segmentation purposes, but many more applications have been seen. So, the potential of this architecture is increasing. Different variants of U-net have been introduced to the world since the first introduction of U-net. We have used six different architecture variants of this model to compare model performance for our domain.

4.2.1 Vanilla U-net

The Vanilla U-net [4] architecture (figure 4.2a) is the base of all the U-net architecture variants used for comparison in this paper. This particular architecture has two paths, one of them is the contracting path on the left side and the other one is the expansive path on the right side. The contracting path is made with convolution blocks (figure 4.2b) or encoder blocks which are shown with yellow tint. The convolution blocks consist of 3×3 convolution layers (padding = "same") according to the layer defined beforehand followed by Batch Normalization and a rectified linear unit (ReLU). After this convolution block, a 2×2 max pooling operation is done with stride 2. This convolution block and max-pooling layer is a repeated application for the down-sampling of this architecture.



(a) Vanilla U-net

(b) Encoder block

(c) Decoder block

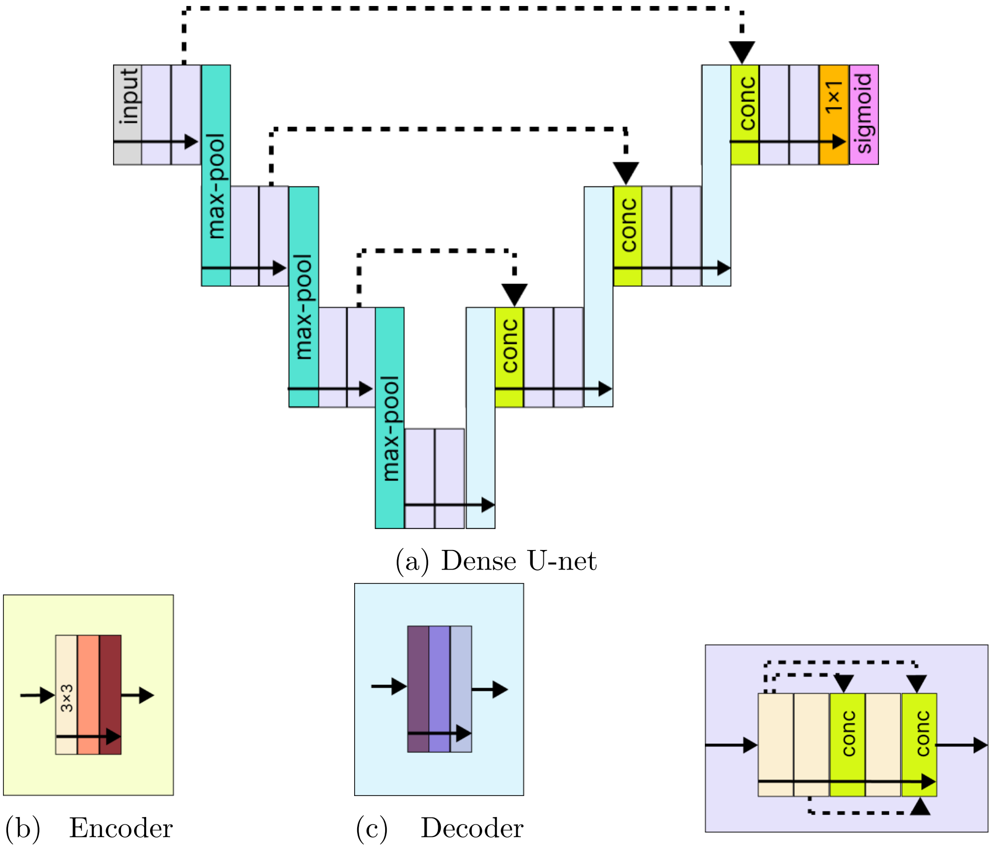
Figure 4.2: Architecture of Vanilla U-net

Furthermore, for the expansive path, it creates the decoder block (figure 4.2c) which is shown in figure as blue tint. This decoder block has 2 × 2 convolution layers that take the feature map from the previous layer as input, followed by Batch Normalization and RelU activation functions. The output of this decoder block then concatenates with the adequately aligned skip connection feature map from the convolution block or the encoder block. Skip connections are a crucial part of this architecture as it provides feature information that gets lost in the deep architecture.

Finally, a 1 × 1 convolution is used on the output of the final layer with a sigmoid activation function having stride 1.

4.2.2 Dense U-net

The Dense-Unet [18] architecture (figure 4.3a) replaces the Convolution and pooling operation with the Dense blocks to have a deep network structure. Functionally, there are two paths in Dense-Unet architecture; the Dense upsampling path (on the left) and the Dense down-sampling path (on the right).



(a) Dense U-net

(b) Encoder block

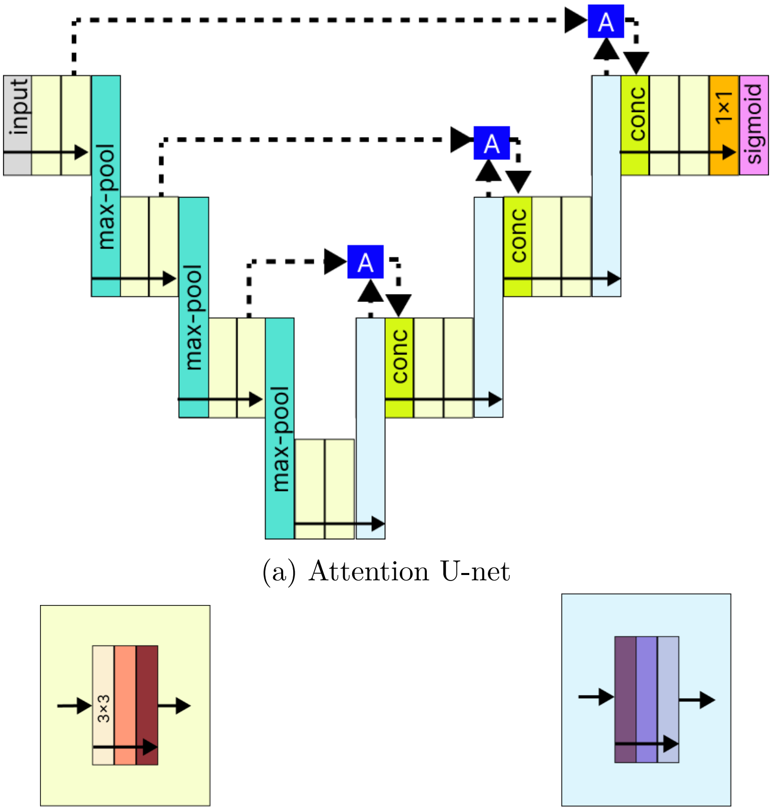
(c) Decoder block

(d) Dense block

Figure 4.3: Architecture of Dense U-net

Instead of a convolution block or encoder block in Vanilla-Unet, Dense-Unet dense block is used here. For the downsampling, there are densely connected convolutional layers in each dense block(figure 4.3d) with batch normalization and ReLU activation function. This dense block is shown with the purple tinted block throughout the architecture. Then, the Transition Block, containing a max pooling layer of 2×2 with a stride of 2, is applied between two Dense blocks to concatenate the dense blocks perfectly. After all the downsampling operation is over, the upsampling operation begins, and each up-sampling layer contains a transpose convolution operation(figure 4.3c) with a stride of 2 and the same padding. There are merge operations after every up-sampling layer to concatenate their extracted features with the adequately aligned skip connection feature map from the dense or encoder block. Then, each merged output is fed into a dense block, and the upsampling operation repeats until there are 4 upsampling layers. Finally, after 4 upsampling layers, a convolution layer performs a 1 × 1 convolution operation with a sigmoid activation function having stride 2, generating the final output.

4.2.3 Attention U-net



(a) Attention U-net

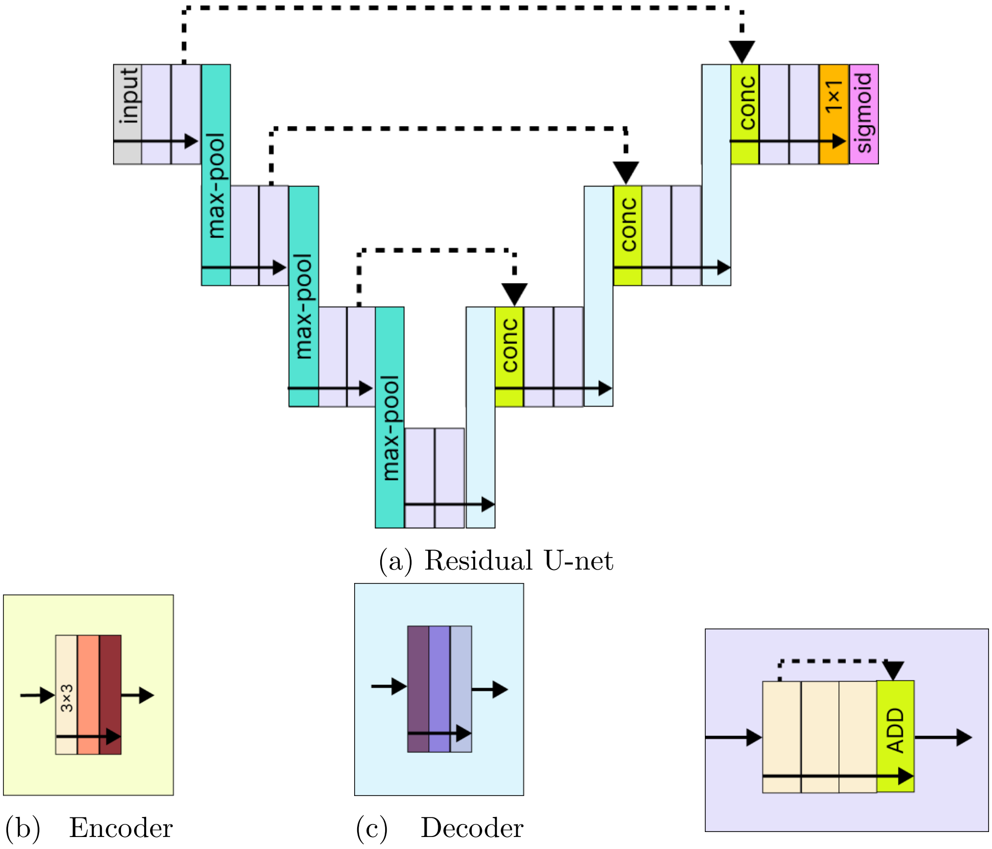
(b) Encoder block

(c) Decoder block

Figure 4.4: Architecture of Attention U-net

An attention U-Net (figure 4.4a) [10] is a variant of the U-Net architecture that incorporates attention mechanisms into the network. During training, attention in Unet is a technique to draw attention to only the relevant activations. It improves network generalization and reduces computing resources wasted on pointless activations. There are two types of attention, Hard Attention, and Soft Attention. We know, Unet skip connection combines spatial information from the down-sampling path with the up-sampling path to retain good spatial information. But this process brings along poor feature representation from the initial layers. A soft attention gate implemented at the skip connections actively suppresses activations at irrelevant regions. This attention gate is shown in figure 4.4a with a square shaped block with A. In this attention block, the gating signal and skip connection go through 1 × 1 convolution having stride 1 (padding= ”valid”), then batch normalization. Then we add them, and the aligned weights get larger while unaligned weights get relatively minor. The added value then goes through the ReLU activation function. After that, comes a 1 × 1 convolution operation with stride 1 (padding= ”valid”) having only 1 filter. Furthermore, these weights are passed through a sigmoid activation function. Finally, this weight multiplies with the element-wise original skip connection, and this output goes to the next layer. This is how the attention block is incorporated in Unet architecture.

4.2.4 Residual U-net



(a) Residual U-net

(b) Encoder block

(c) Decoder block

(d) Residual block

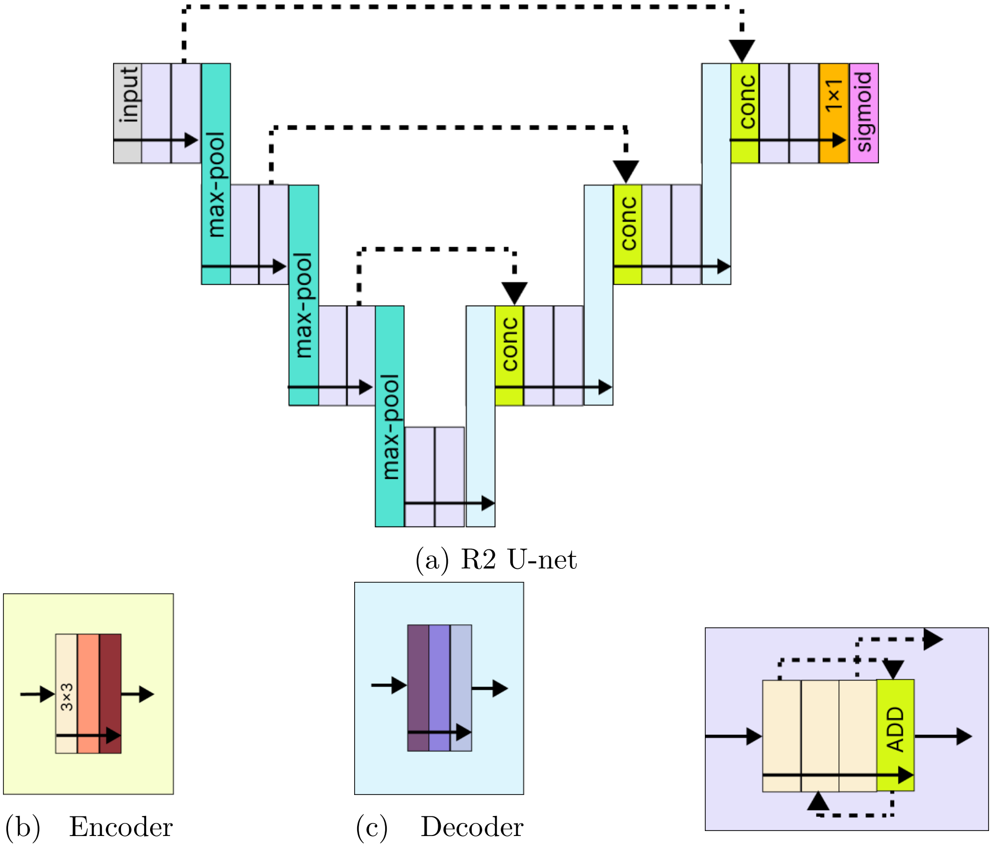
Figure 4.5: Architecture of Residual U-net

A residual U-Net (figure 4.5a) [6] is a variant of the U-Net architecture that incorporates residual connections into the network. Residual connections are a type of skip connection that allows the network to learn residual functions or the difference between the input and the desired output of a layer. Residual networks overcome the problems of deep convolutional neural networks. Stacking convolutional layers and making the model deeper hurts the network’s generalization ability. Residual network architecture was introduced, which adds the idea of ”skip connections”, addressing this problem. In traditional neural networks, each layer feeds into the next layer. In networks with residual blocks, each layer provides into the next layer and directly into the layers about 23 hops away. Inputs can propagate faster through the residual connections (shortcuts) across layers.

In our case, Residual blocks (figure 4.5d) are implemented in place of convolution or encoder blocks of unet architecture as shown with purple block and create a Residual-Unet model. Residual blocks consist of 3 × 3 convolutional layers with stride 1 (padding = ”same”), batch normalization and ReLU activation function. After calculating one set of convolutional operations, that value adds to the last layer output as the residual connection. After these four downsampling operations, the upsampling process begins. The previous residual blocks output of the downsampling acts as an input for the deconvolution block where a Conv2DTranspose procedure is done with stride 2 (padding = ”same”). This value feeds into the batch normalization and ReLU activation function. The returned value of this deconvolution block then concatenates with the element-wise original skip connection, and this output goes through the residual block and this process repeats for all the element-wise operations. Finally, the output of the last upsampling block is passed through a 1 × 1 convolution operation with a sigmoid activation function and generates the outcome.

4.2.5 R2 U-net

R2U-Net (Residual and Recurrent U-Net) (figure 4.6a) is a variant of the U-Net architecture that incorporates both residual connections and recurrent connections into the network. According to [7], some of the benefits of this network architecture is that while we train the deep network architecture, a residual unit helps. Another advantage is that segmentation tasks get better feature map accumulation with recurrent residual convolutional layers, which ensures better feature representation. Lastly, this allows us to have a better version of Unet architecture with the same network parameters. For our research, we again use RRCNN(Recurrent Residual convolutional neural network) blocks (figure 4.6d) instead of convolutional blocks or encoder blocks as shown with purple tint. The input first goes through this RRCNN block, and one 3 × 3 convolution operation with stride 1 (padding = ”same”) is done on the given input then recurrent unit implements two recurrences for the convolutional layers with batch normalization and ReLU activation function. Then the output of the initial convolution is added to the recurrent unit’s output, and this final value is the output of the RRCNN block. The previous RRCNN blocks output of the downsampling acts as an input for the deconvolution block where a Conv2DTranspose procedure is done with stride 2 (padding = ”same”). This value feeds into the batch normalization and ReLU activation function. The returned value of this deconvolution block then concatenates with the element-wise original skip connection, and this output goes through the RRCNN block and this process repeats for all the element-wise operations. After the four upsampling operations, finally, with 1 × 1 convolution operation with sigmoid activation function, we get the outcome.



(a) R2 U-net

(b) Encoder block

(c) Decoder block

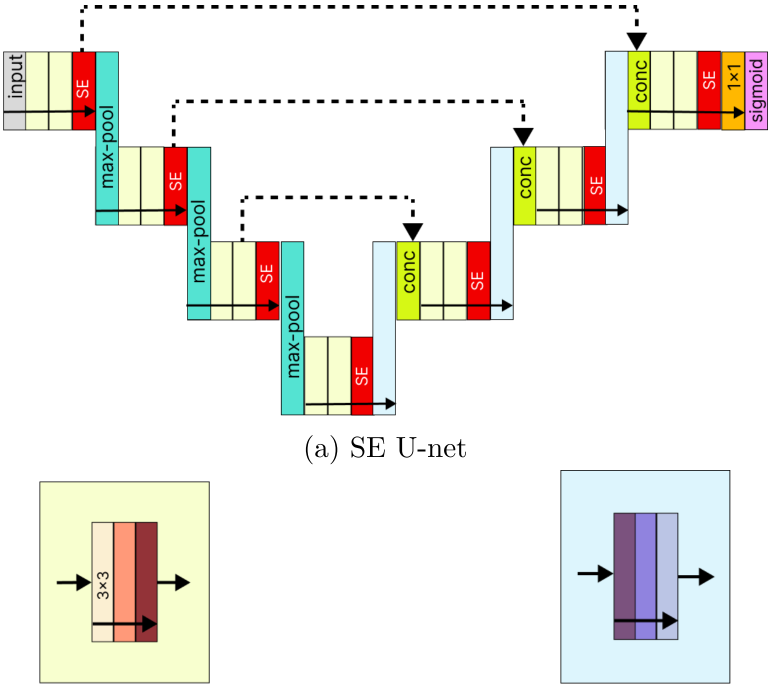
(d) R2 block

Figure 4.6: Architecture of R2 U-net

4.2.6 SE U-net

SE-Unet (figure 4.7a) [11] is a variant of the U-Net architecture that incorporates squeeze-and-excitation (SE) blocks into the network. SE blocks are a type of attention mechanism that allows the network to dynamically weigh the features at each position in the input data based on their relevance to the task at hand, which can be helpful for tasks such as image segmentation, where the network needs to accurately identify and segment objects in the image, even if they are small or partially occluded. To implement SE blocks in a U-Net, they are typically added after each convolutional layer in the network. Each SE block consists of two parts: a squeeze layer, which reduces the dimensions of the input data by aggregating the features along the channel dimension, and an excitation layer, which re-weights the features using a learnable weighting function. The excitation layer allows the network to assign higher weights to essential features in the input data and lower weights to less essential features, which helps the network to focus on the most relevant features when making predictions.

In our research, we use Scse-blocks as the red marked block in figure 4.7a, after each encoder block and decoder block, and this Scse-block takes input from the encoder block. Scse-block consists of Cse and Sse-blocks. The Sse-block takes input and runs a $1x$ convolutional operation with a sigmoid activation function with only one filter(use_bias = ”False”). After that, this value is multiplied by the input and returned to the Scse-block. For the Cse-block, it first performs spatial squeeze with



(a) SE U-net

(b) Encoder block

(c) Decoder block

Figure 4.7: Architecture of SE U-net

global average pooling. This value is used to perform two fully connected convolutional layers. One has a ReLU activation function, and the other has a sigmoid activation function. The output from this then excites the input feature map. Furthermore, Scse-block gets the result from Cse-block. These two blocks' value is then compared, and their maximum is returned as the value of Scse-block. Finally, the output of the Scse-block is passed on to the next encoder or decoder block. In the end, like any other unet model in the last layer, a 1×1 convolution is performed with a sigmoid activation function to generate the output.

4.3 Implementation of Architectures

We aim to perform semantic segmentation on dental x-ray images (panoramic x-ray images). To achieve that goal, we are training six different unet variants.

4.3.1 Training and Architecture Parameters

Our network architectures use a 1×1 convolution layer with stride one followed by a sigmoid activation. The output of these networks gives us binary classification probabilities corresponding to each pixel in the original input teeth x-ray images. All the networks use 2×2 max-pooling and transposed convolution on the downsampling and upsampling, respectively. Furthermore, Network architecture consists of four downsampling and four upsampling layers. The first downsampling layer starts with 16 filters, and as the network goes deep, the filter size increases twice the previous amount. However, the filter size decreases by half the prior amount for the upsampling layers. We use 2 and 3 convolution operations per block for performance comparison for each of these encoder and decoder blocks. These convolution operations are followed by batch normalization and ReLU activation functions.

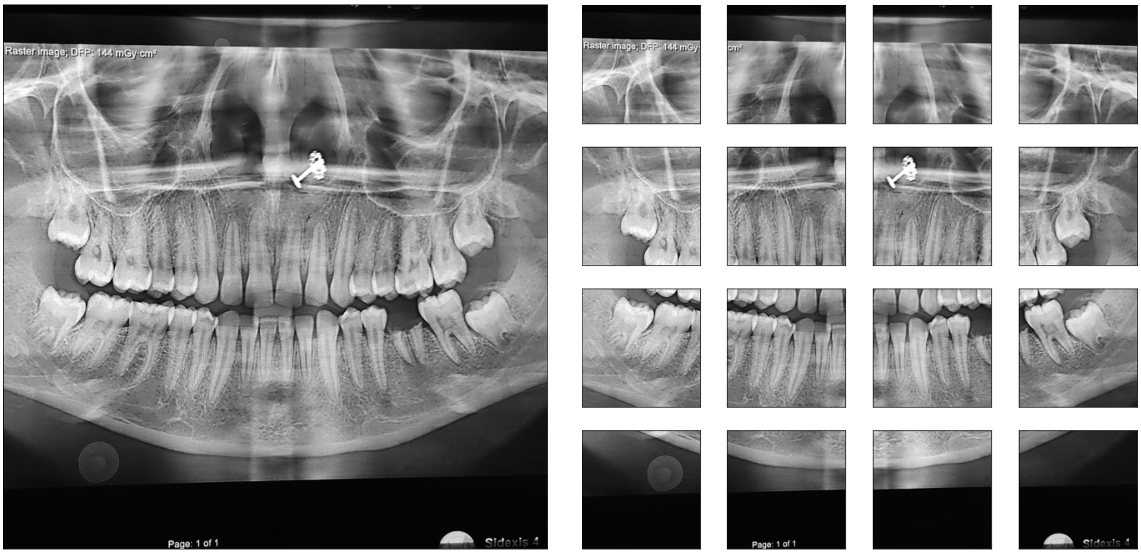
For training purposes, adam optimizer is used with a learning rate of 0.0001, the batch size is 16, and all the models are trained for 100 epochs. The dice_coef_loss function is used to calculate the loss, and for accuracy metric, we are using "accuracy", "dice-coefficient", "f1-score", and "Iou" on the validation set. To compare the networks, we consider these metrics with the best epoch.

4.3.2 Semantic Segmentation on Patches

After training all of our architectures, we have predicted on some test images using all of the models and each of their variants. The steps are similar for all architectures.

As shown in figure 4.8, first, we need to create patches out of the image that we want to predict. To do so, we will take a dental x-ray image as our input that we would like to segment i.e., predict using our model (figure 4.8a).

Next, we divide our image into 4 × 4 = 16 patches, each with a 256 × 256 pixel size. Also, we need to normalize all the values before feeding the patches for prediction (figure 4.8b).



(a) Original image (input)

(b) Created patches from the image

Figure 4.8: Creating patches

As we have already trained our model, we can use it to predict and segment an image. So, now we will use our model to predict each of the patch (figure 4.9).

After predicting all the patches, we will be left with 16 segmented patches for an image.

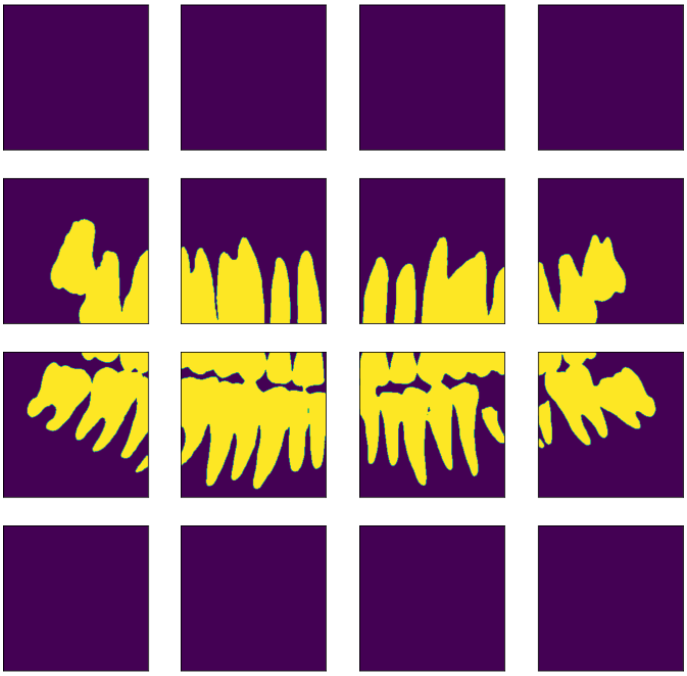
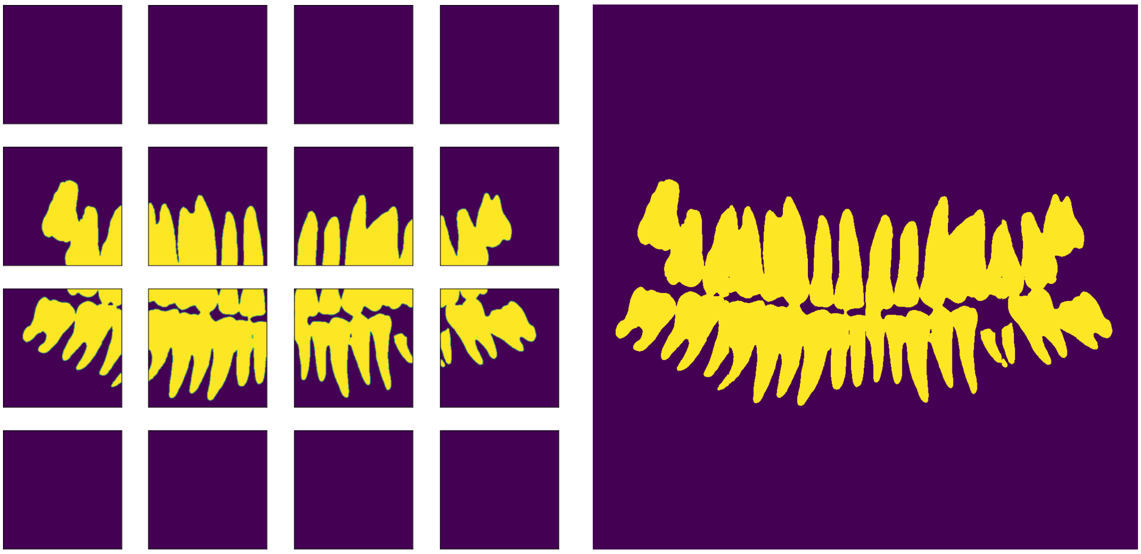


Figure 4.9: Segmentation on patches

4.3.3 Reconstruction of Segmented Image

As illustrated in figure 4.10, after we get the 16 segmented patches, we will reconstruct the entire image. In simpler words, we will put together all 16 segmented patches and form the entire image with a final size of 1024 × 1024 pixels.



(a) Segmented patches

(b) Reconstruction of the patches

Figure 4.10: Reconstruction of the segmented image

4.4 Evaluation Method

As we aim to train segmentation models, more than common metrics such as accuracy or f1 score is needed to evaluate our models properly. Alongside these metrics, we have considered our two-layer and three-layer variants on dice coefficient and IoU for further clarification by comparing the training and validation curves. Furthermore, we used a confusion matrix as a tabular analysis to evaluate the performance of our U-net variants.

4.4.1 Performance Metrics

We have used accuracy, Dice coefficient (Dice's similarity coefficient), F1 score, and Intersection over Union (IoU) to evaluate the performance of our binary image segmentation models. By using these four metrics to evaluate our models' performance, we can better understand how well the models perform on the image segmentation task.

The four quantities TP, FP, TN, and FN are used to calculate various performance metrics for binary image segmentation, including accuracy, Dice coefficient, IoU, etc. In binary image segmentation, true positives (TP) are the number of pixels that are correctly classified as foreground (teeth). False positives (FP) are the number of pixels that are incorrectly classified as foreground (i.e., they should be the background). True negatives (TN) are the number of pixels correctly classified as background. False negatives (FN) are the number of pixels incorrectly classified as background (i.e., they should be foreground).

Accuracy

In binary image segmentation, the accuracy metric can be defined as the number of pixels that are correctly classified by the model divided by the total number of pixels in the image. More formally, the accuracy can be calculated using the following formula:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{4.1}$$

This formula can be used to calculate the accuracy of a binary image segmentation model, where the model is trying to classify each pixel in the image as either belonging to the foreground or the background.

Vanilla U-net In figure 4.11, it can be seen that the train accuracy and validation accuracy are both increasing over time, and the difference between the two is small and stable for the Vanilla U-net (two-layer variant) architecture.

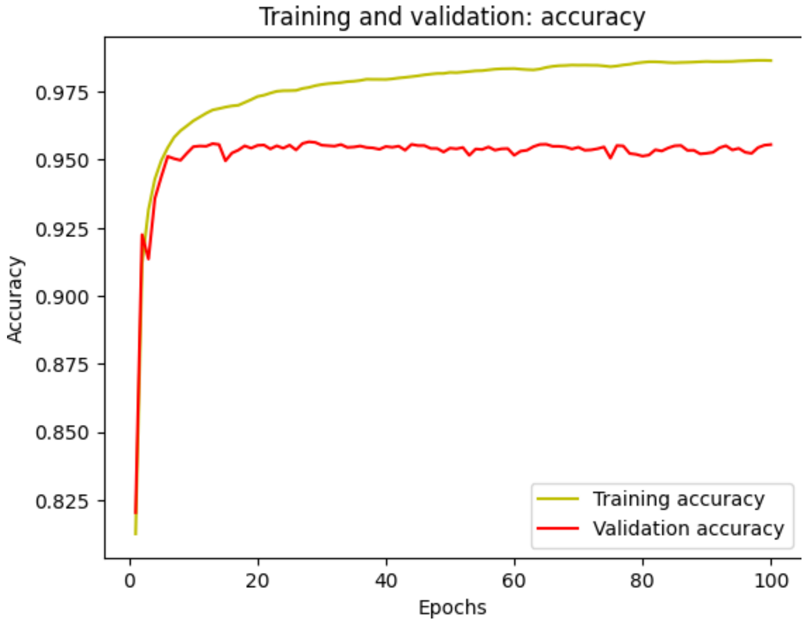


Figure 4.11: Validation accuracy for Vanilla U-net (two-layer variant)

Correspondingly, figure 4.12 for three-layer variant Vanilla U-net illustrates similar result. The difference between the training and validation accuracy is typically small, which indicate that the model is generalizing well and is not overfitting to the training data, this is a good sign for the model's performance.

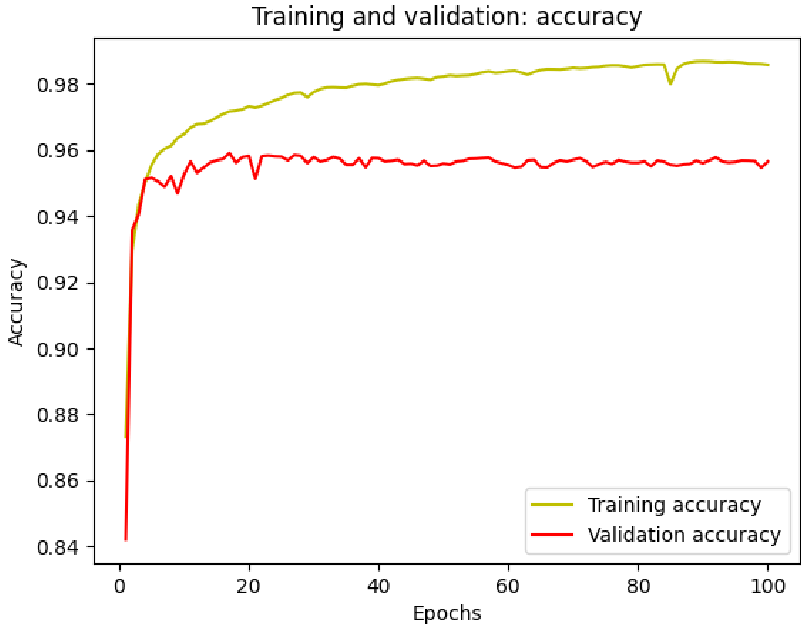


Figure 4.12: Validation accuracy for Vanilla U-net (three-layer variant)

Attention U-net In figure 4.13, we can see that the training accuracy increases as the model is trained and reaches a high value for the Attention U-net model with two convolutional layers. The validation accuracy increases as the model are trained, but plateau at a lower value than the training accuracy and stays relatively stable. This suggests that the model is balanced with the training data.

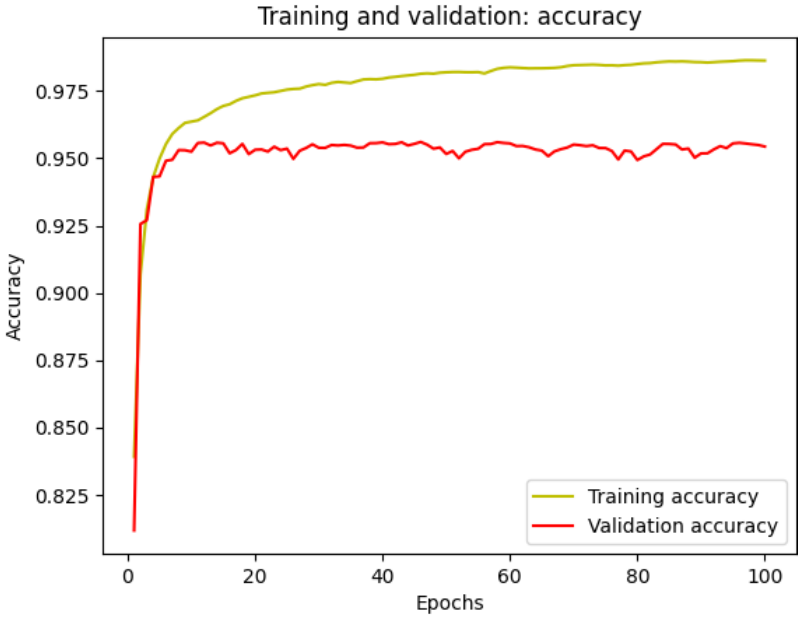


Figure 4.13: Validation accuracy for Attention U-net (two-layer variant)

Likewise, looking at figure 4.14, we can draw similar conclusion. However, with addition of one convolutional layers per block, the gap between the training accuracy and validation accuracy is narrower compared to the previous architecture.

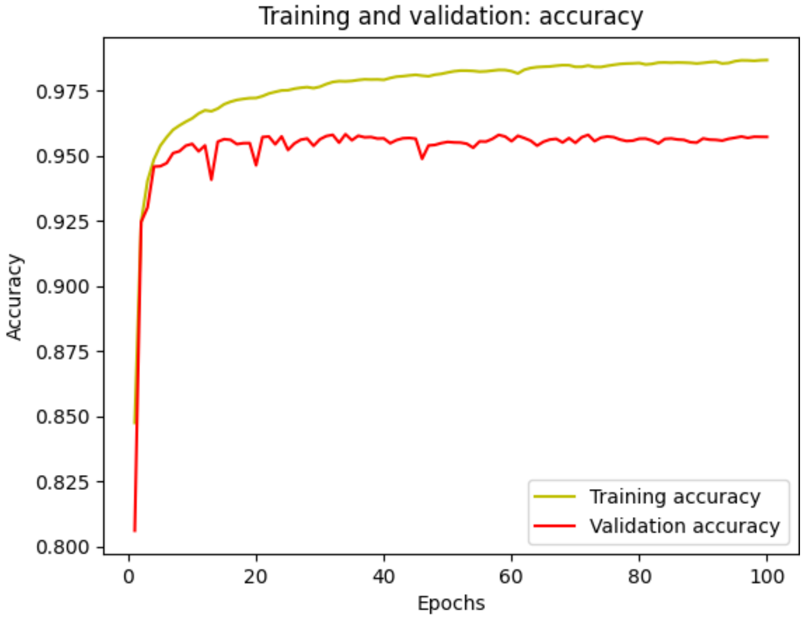


Figure 4.14: Validation accuracy for Attention U-net (three-layer variant)

Dense U-net As seen in figure 4.15, the model's performance, Dense Unet (two-layer variant), seems decent as the validation accuracy is close to training accuracy. This could be a sign that the model can learn the general pattern of image segmentation from the training data and apply the same to the validation data.

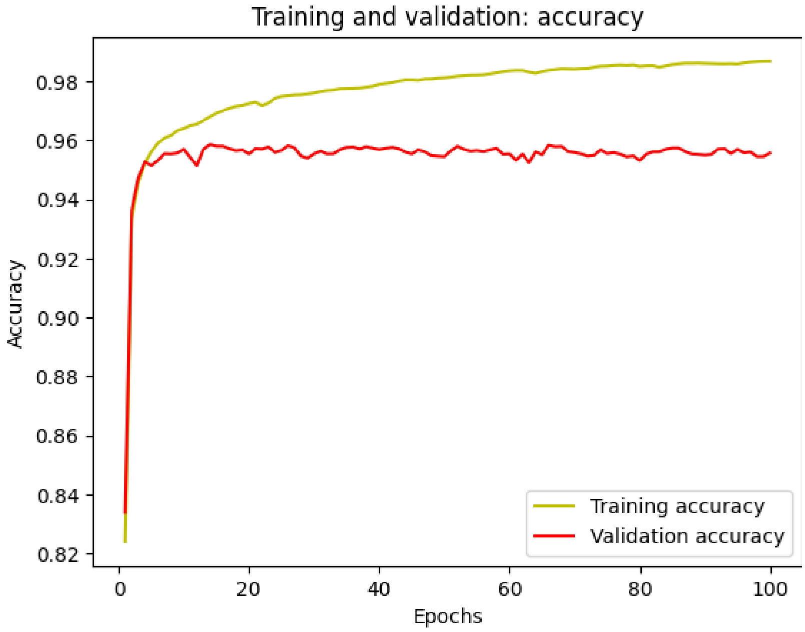


Figure 4.15: Validation accuracy for Dense U-net (two-layer variant)

Similarly, figure 4.16 depicts similar results for this model by adding one convolutional layer per block. The difference between the training and validation accuracy is typically tiny, indicating that the model is generalizing well and is balanced with the training data.

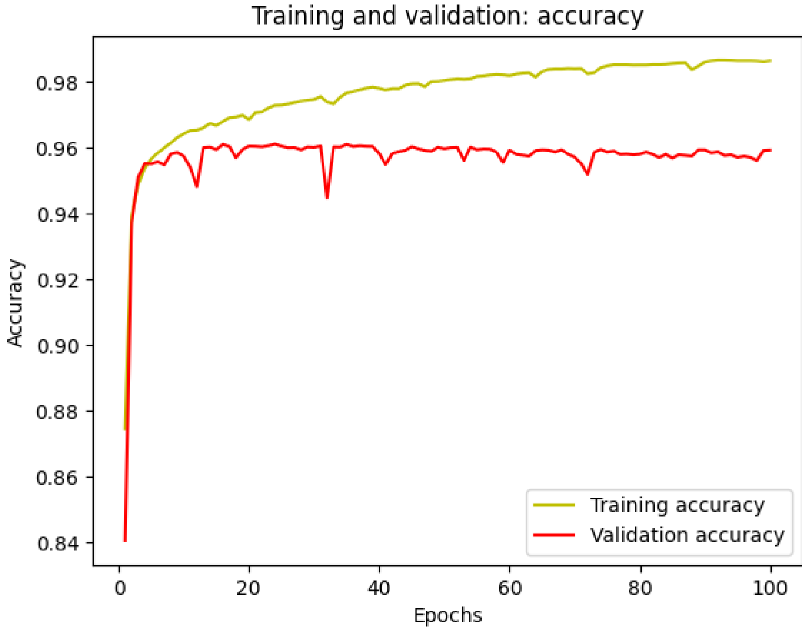


Figure 4.16: Validation accuracy for Dense U-net (three-layer variant)

R2 U-net We can see from figure 4.17 for R2 U-net (two-layer variant) that the validation accuracy is stable and close to training accuracy. This is a positive indication of the model's performance on unseen data.

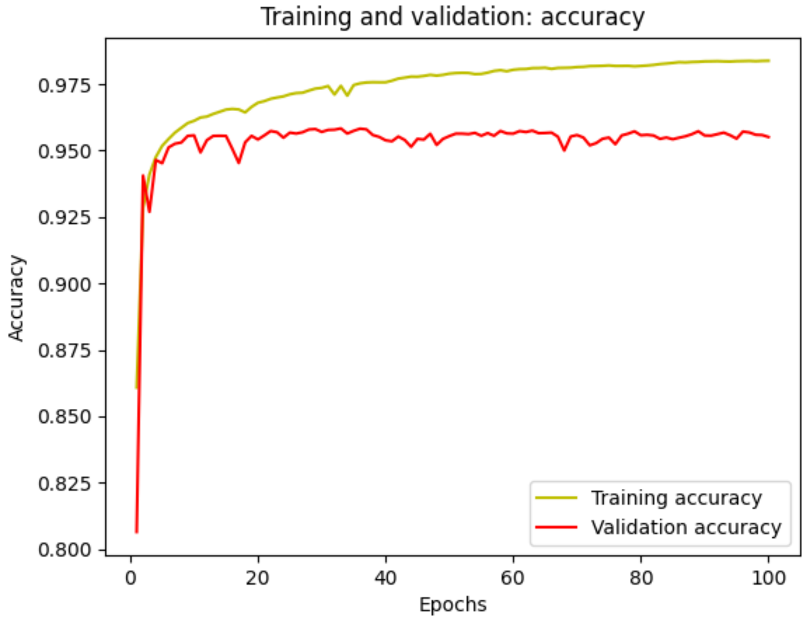


Figure 4.17: Validation accuracy for R2 U-net (two-layer variant)

As seen in figure 4.18 for R2 U-net (three-layer variant), the validation accuracy fluctuates a bit in the early stages of training which is typical. However, the validation accuracy is stable after a certain point, suggesting the model is not overfitting. Even the gap between training and validation accuracy is even narrower than the two-layer variant of this model.

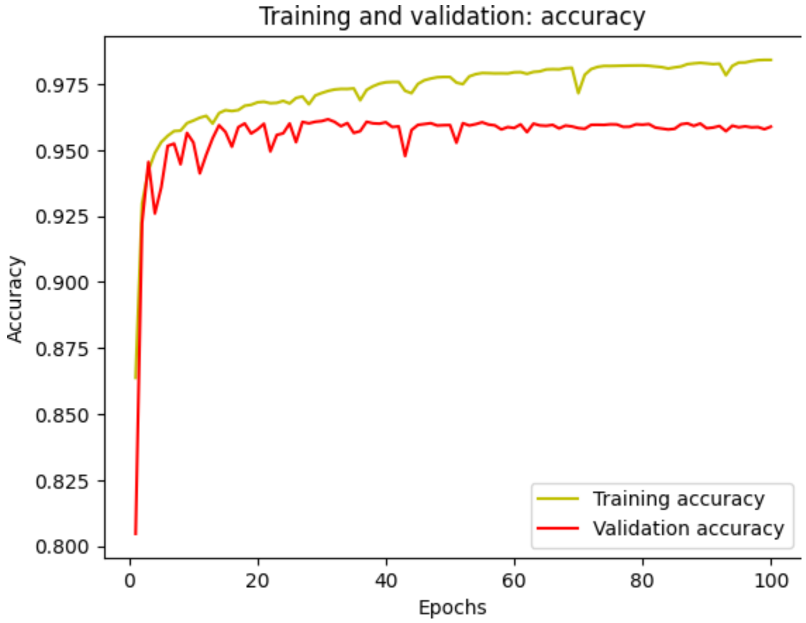


Figure 4.18: Validation accuracy for R2 U-net (three-layer variant)

Residual U-net Based on figure 4.19, Residual U-net (two-layer variant) is well-performing in terms of accuracy and generalization. The model's performance seems robust as training and validation accuracy has reached a high value. Also, the validation accuracy is stable after a certain point which suggests the model is not overfitting.

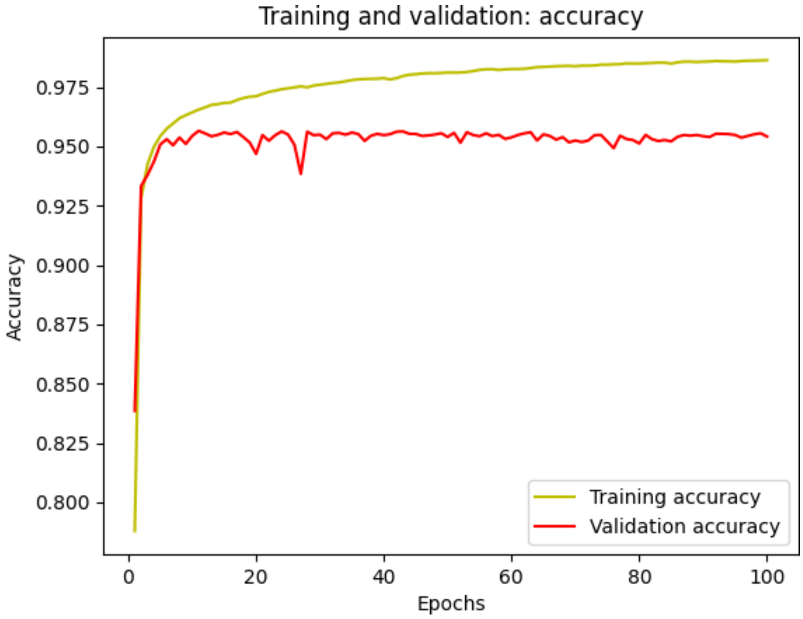


Figure 4.19: Validation accuracy for Residual U-net (two-layer variant)

Similarly, figure 4.20 illustrates that the gap between the training and validation accuracy is low and even narrower than the model previously mentioned. It usually depicts that this model, Residual U-net (with three convolutional layers), fits the training data well and can generalize to unseen data. This indicates that the model will perform well on new, unseen data.

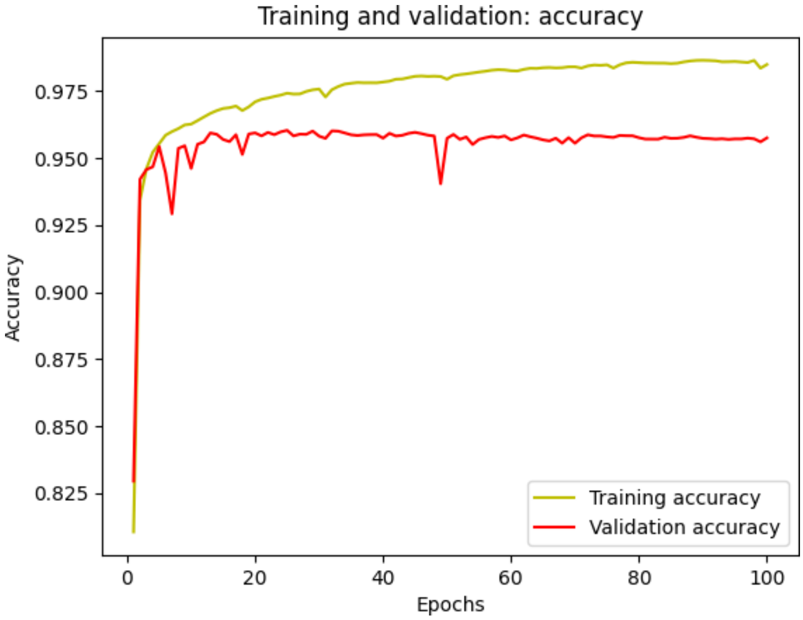


Figure 4.20: Validation accuracy for Residual U-net (three-layer variant)

SE U-net As seen in figure 4.21, the training accuracy and validation accuracy are relatively close together throughout the entire training process, with the validation accuracy staying slightly behind the training accuracy. This suggests that the two-layer variant SE U-net model is generalizing well and is balanced with the training data.

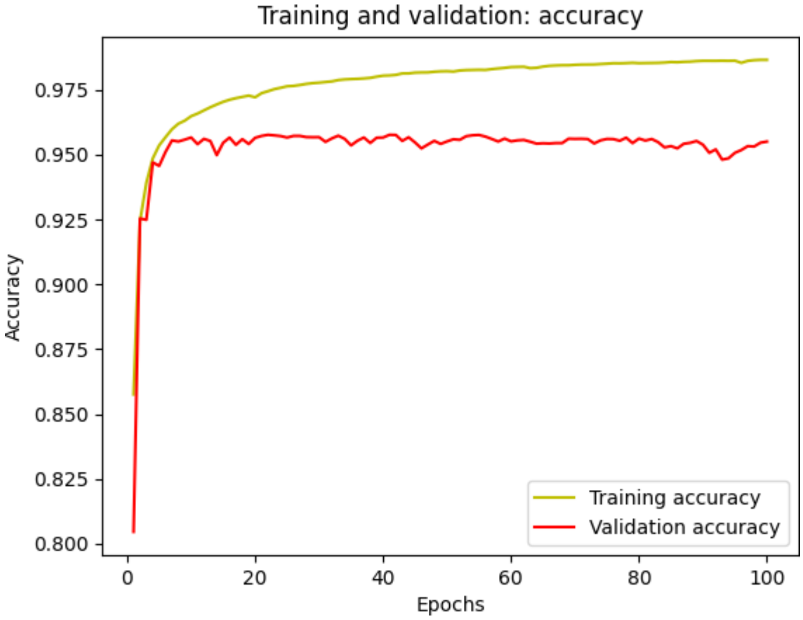


Figure 4.21: Validation accuracy for SE U-net (two-layer variant)

Likewise, figure 4.22 indicates that with the addition of another convolution layer, this model can perform well on both the training and validation sets and is likely to perform well on unseen data. Based on this graph, the model is well-performing in terms of accuracy and generalization.

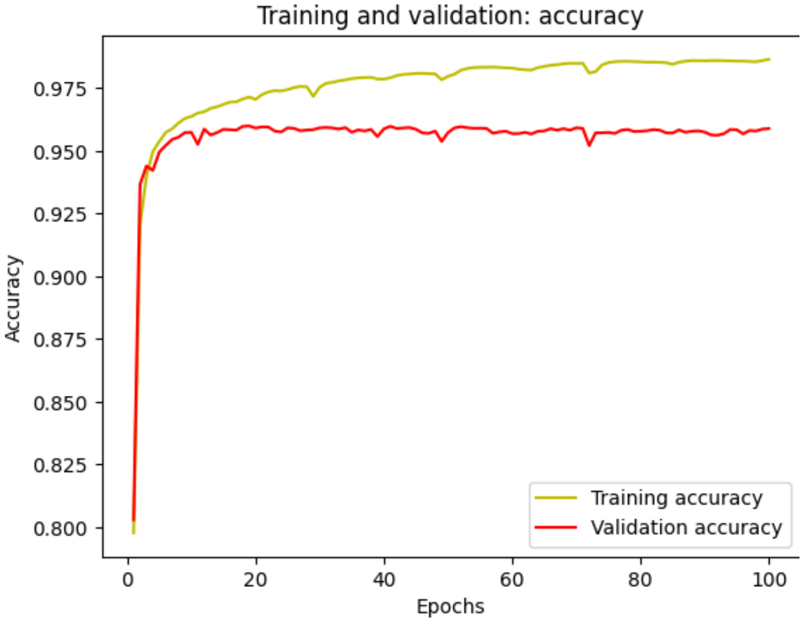


Figure 4.22: Validation accuracy for SE U-net (three-layer variant)

F1 score

It is worth noting that accuracy is a simple metric and may only sometimes be the best choice for evaluating the performance of a binary image segmentation model. For segmentation tasks, the F1 score can be a better performance metric than accuracy in some cases. The main reason is that while accuracy measures the proportion of correctly classified pixels, it does not consider false positives and negatives, which can be critical in segmentation tasks. A model with high accuracy may still need to catch more of the structure of interest, which could have severe consequences in specific applications such as medical imaging.

The F1 score, on the other hand, represents a harmonic mean of recall and accuracy. The proportion of true positive pixels in the predicted positive pixels is measured by precision. In contrast, the proportion of true positive pixels in the actual positive pixels is measured by a recall. F1 score combines precision and recall in a single metric and balances them, which is particularly useful when it is essential to balance between detecting as many of the structures of interest as possible (recall) and detecting as few false positives as possible (precision).

In the case of teeth segmentation, a model with high precision but low recall may miss many teeth, but the ones it detects will be accurate. This can be a problem in cases where missing teeth can cause serious consequences. While a model with high recall but low precision would detect many teeth, most of them will be false positives. This will cause a lot of noise and confusion and may result in unnecessary treatments. Therefore, the F1 score can be a better performance metric than accuracy, as it considers both precision and recall. It can balance both to comprehensively evaluate the model's performance in detecting teeth.

The equation 4.2 for calculating F1 score is:

$$\text{F1 score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \tag{4.2}$$

The equation 4.3 for calculating precision is:

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{4.3}$$

The equation 4.4 for calculating recall is:

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4.4}$$

Vanilla U-net In figure 4.23, it can be seen that the train f1 score and validation f1 score are both increasing over time, and the difference between the two is small and stable for the Vanilla U-net (two-layer variant) architecture.

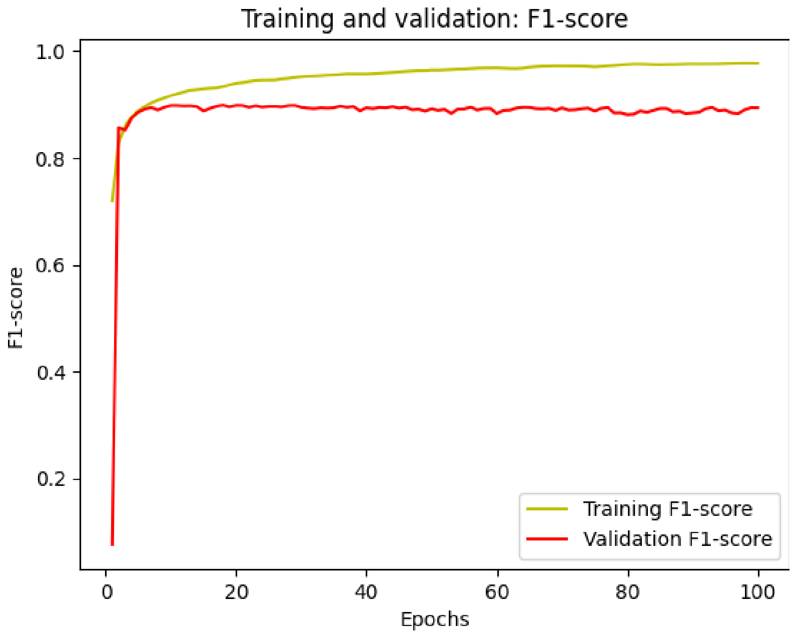


Figure 4.23: Validation f1 score for Vanilla U-net (two-layer variant)

Correspondingly, figure 4.24 for the three-layer variant Vanilla U-net illustrates a similar result. The difference between the training and validation f1 score is typically tiny, which indicates that the model is generalizing well and is not overfitting to the training data; this is a good sign for the model's performance.

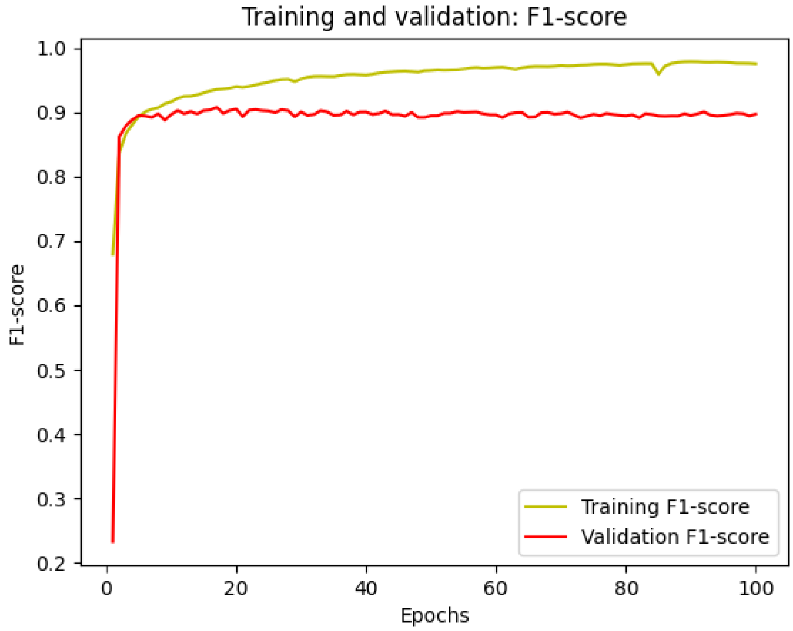


Figure 4.24: Validation f1 score for Vanilla U-net (three-layer variant)

Attention U-net In figure 4.25, we can see that the training f1 score increases as the model is trained and reaches a high value for the Attention U-net model with two convolutional layers. The validation f1 score also increases as the model is trained but plateaus at a lower value than the training f1 score and stays relatively stable. This suggests that the model is balanced with the training data.

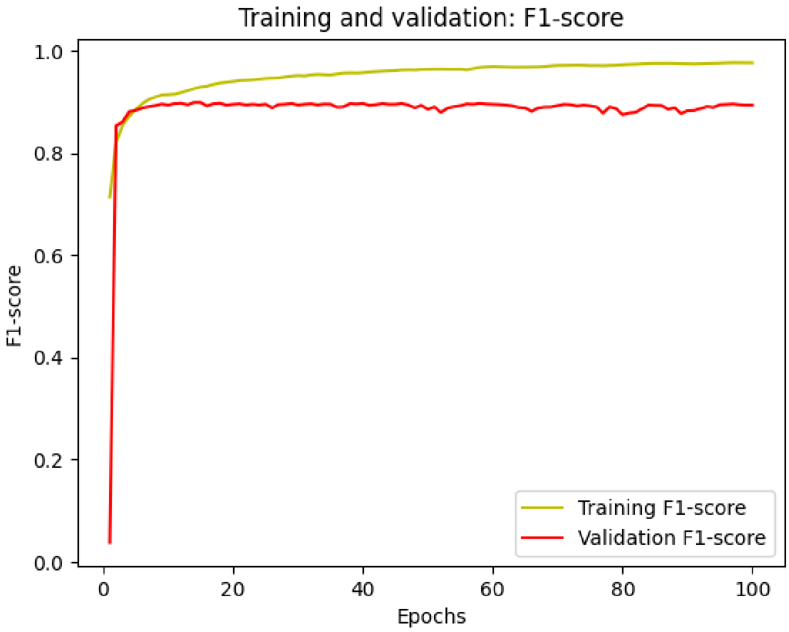


Figure 4.25: Validation f1 score for Attention U-net (two-layer variant)

Likewise, looking at figure 4.26, we can draw a similar conclusion. However, with the addition of one convolutional layer per block, the gap between the training f1 score and validation f1 score is narrower compared to the previous architecture.

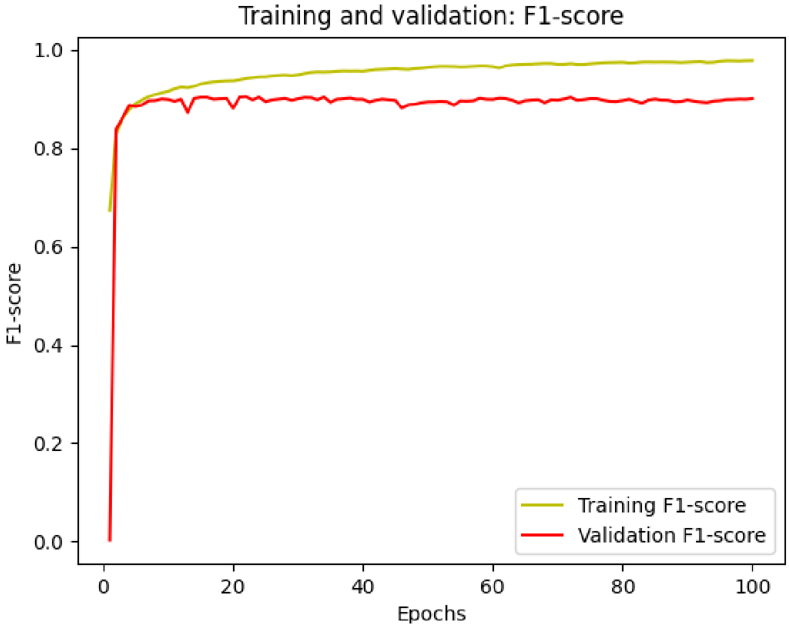


Figure 4.26: Validation f1 score for Attention U-net (three-layer variant)

Dense U-net As seen in figure 4.27, the performance of the model, Dense U-net (two-layer variant), seems decent as the validation f1 score is close to the training f1 score. This could be a sign that the model is able to learn the general pattern of image segmentation from the training data and applies the same to the validation data as well.

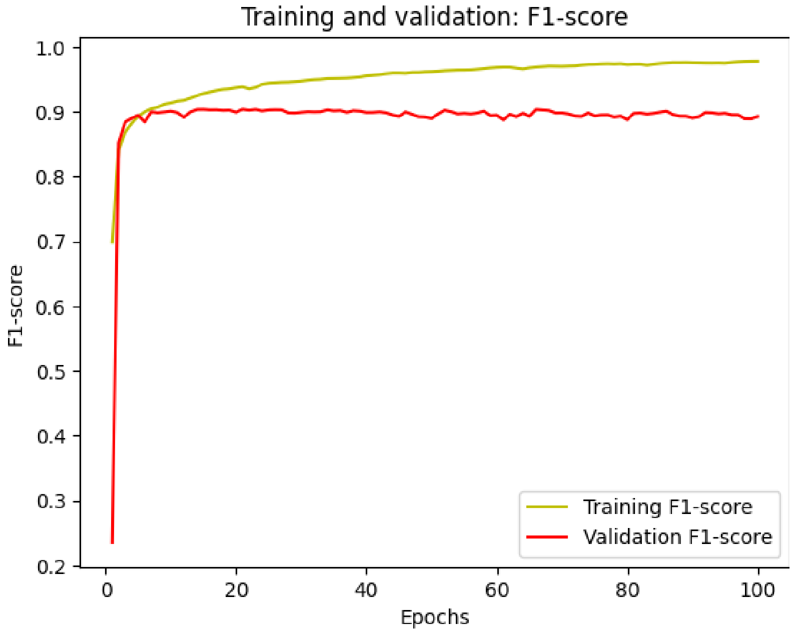


Figure 4.27: Validation f1 score for Dense U-net (two-layer variant)

Similarly, figure 4.28, depicts similar results for this model by adding one convolutional layer per block. The difference between the training and validation f1 score is typically small, indicating that the model is generalizing well and is not overfitting to the training data.

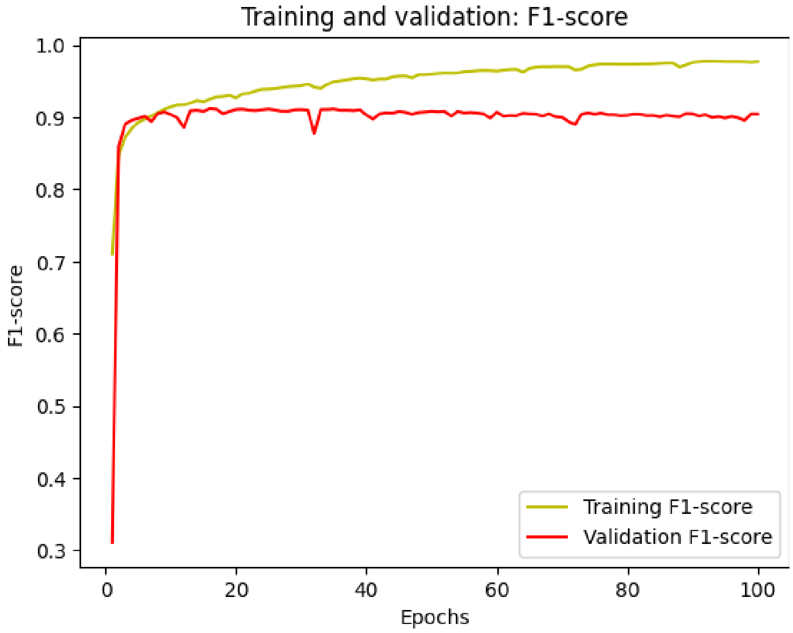


Figure 4.28: Validation f1 score for Dense U-net (three-layer variant)

R2 U-net We can see from figure 4.29 for R2 U-net (two-layer variant) that the validation f1 score is stable and close to the training f1 score. This is a positive indication of the model's performance on unseen data.

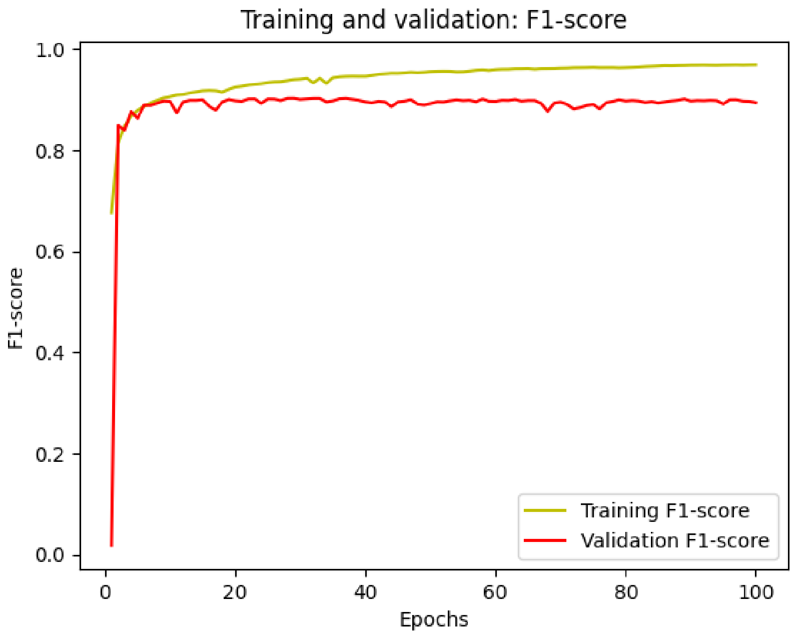


Figure 4.29: Validation f1 score for R2 U-net (two-layer variant)

As seen in figure 4.30 for R2 U-net (three-layer variant), the validation f1 score appears to be fluctuating a bit in the early stages of training which is typical. However, the validation f1 score is stable after a certain point which suggests the model is not overfitting. Even the gap between the training f1 score and validation f1 score seems to be even narrower than the two-layer variant of this model.

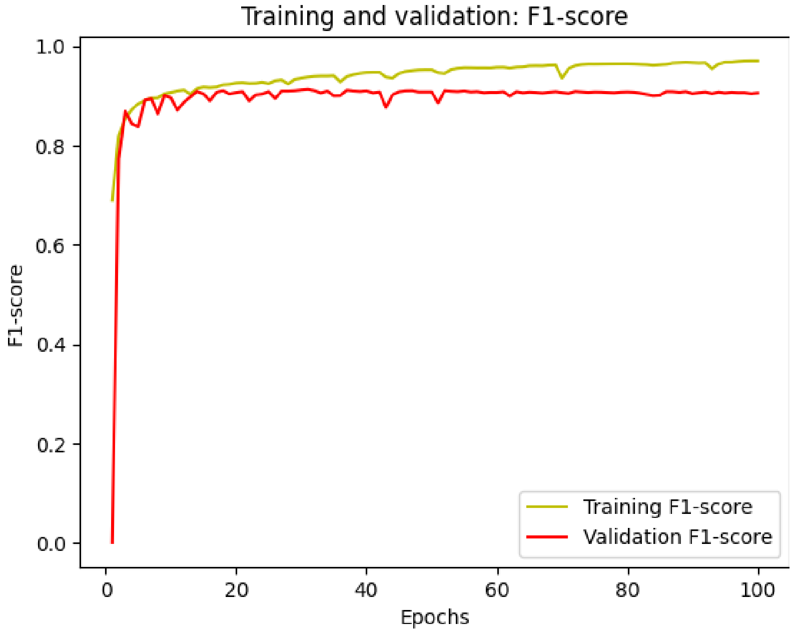


Figure 4.30: Validation f1 score for R2 U-net (three-layer variant)

Residual U-net Based on figure 4.31, it appears that Residual U-net (two-layer variant) is well-performing in terms of accuracy and generalization. The model's performance seems robust as both training and validation f1 score have reached a high value. Also, the validation f1 score is stable after a certain point which suggests the model is not overfitting.

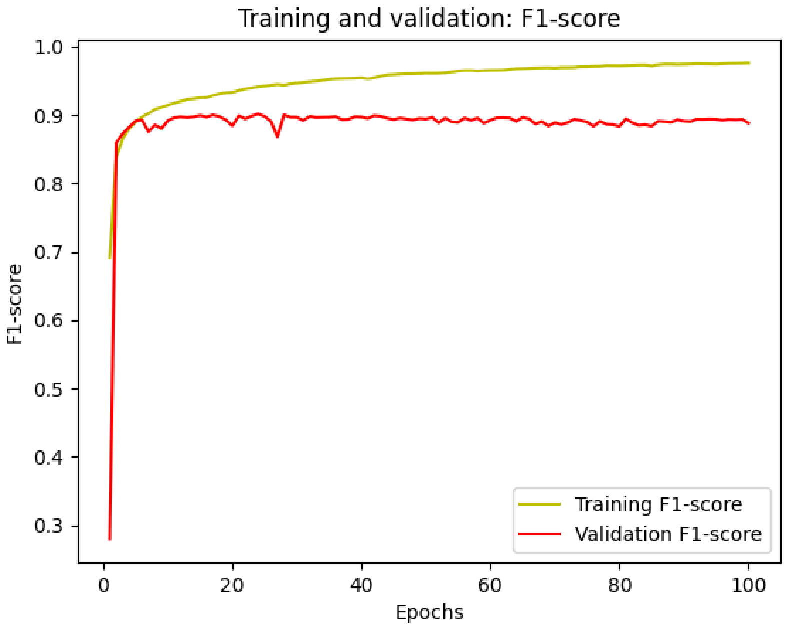


Figure 4.31: Validation f1 score for Residual U-net (two-layer variant)

Similarly, figure 4.32 illustrates that the gap between the training and validation f1 score is low and even narrower than the model previously mentioned. It usually depicts that this model, Residual U-net (with three convolutional layers), is not only fitting the training data well but also able to generalize to unseen data. This is a good indication that the model will perform well on new, unseen data.

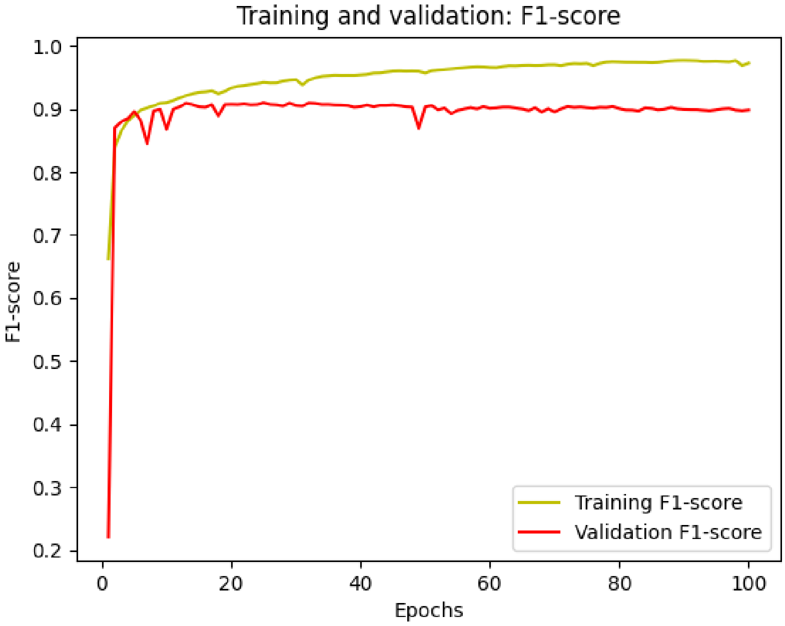


Figure 4.32: Validation f1 score for Residual U-net (three-layer variant)

SE U-net As seen in figure 4.33, the training f1 score and validation f1 score are relatively close throughout the training process, with the validation f1 score staying slightly behind the training f1 score. This suggests that the model, the two-layer variant SE U-net, is generalizing well and is not overfitting to the training data.

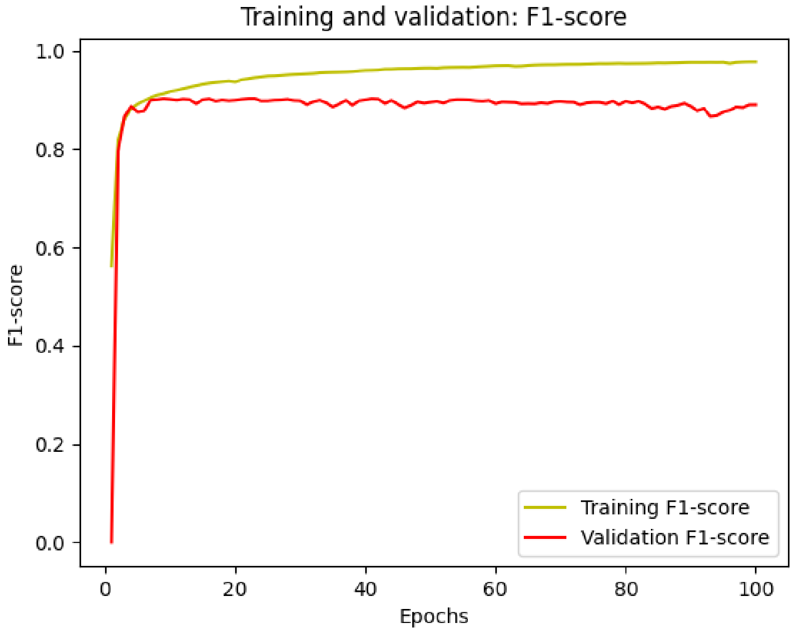


Figure 4.33: Validation f1 score for SE U-net (two-layer variant)

Likewise, figure 4.34 indicates that this model, with the addition of another convolution layer, is able to perform well on both the training and validation sets and is likely to have a good performance on unseen data. Based on this graph, it appears that the model is well-performing in terms of accuracy and generalization.

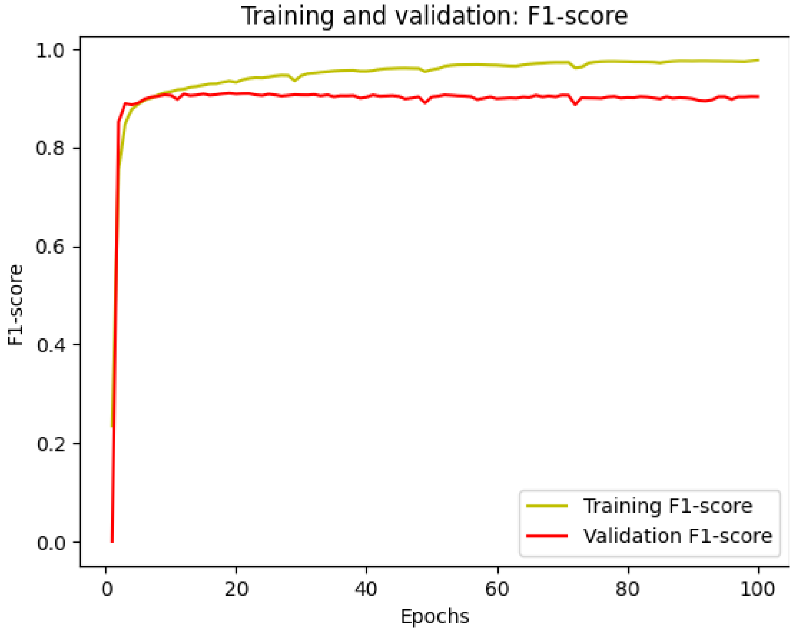


Figure 4.34: Validation f1 score for SE U-net (three-layer variant)

Dice coefficient

A standard metric for assessing how well a segmentation model is working is the dice coefficient. It is frequently used to assess how well models perform in recognizing certain structures or regions of interest in image segmentation, particularly in medical imaging.

As illustrated in figure 4.35, the Dice coefficient (Dice's similarity coefficient) is a metric that measures the overlap between the predicted segmentation and the ground truth segmentation in binary image segmentation. The Dice coefficient is calculated by first finding the intersection between the predicted foreground and the ground truth foreground, and then dividing this value by the sum of the number of pixels in the predicted foreground and the number of pixels in the ground truth foreground.

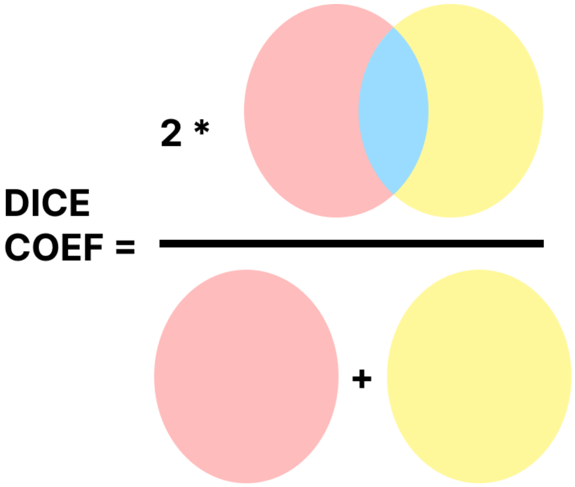


Figure 4.35: Dice coefficient Venn diagram

In terms of teeth segmentation, the model's predicted segmentation set (A) would represent the pixels that the model has identified as teeth, and the ground truth set (B) would represent the pixels that have been manually annotated as teeth. To calculate the true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) from the Dice coefficient sets, we need to compare the predicted segmentation set (A) with the ground truth set (B). The true positives (TP) are the pixels that the model correctly identifies as teeth and are also present in the ground truth set (A   B). The true negatives (TN) are the pixels that the model correctly identifies as background and are not present in the ground truth set. The false positives (FP) are the pixels that the model incorrectly identifies as teeth and are not present in the ground truth set (A-B). The false negatives (FN) are the pixels that the model incorrectly identifies as background and are present in the ground truth set (B-A).

The following equation 4.5 defines dice coefficient:

$$\text{Dice coefficient} = \frac{2 \times \text{TP}}{2 \times \text{TP} + \text{FP} + \text{FN}} \tag{4.5}$$

Vanilla U-net In figure 4.36, it can be seen that the train dice coefficient and validation dice coefficient are both increasing over time, and the difference between the two is small and stable for the Vanilla U-net (two-layer variant) architecture.

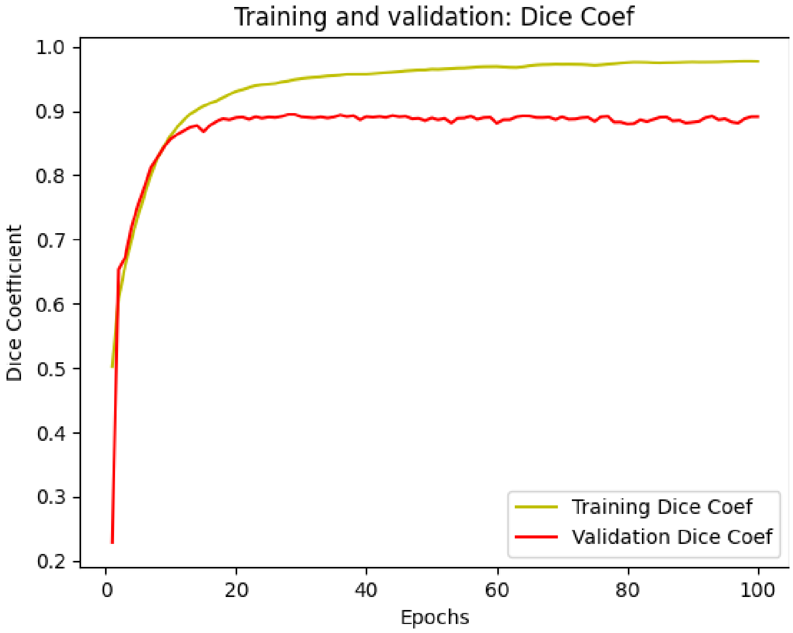


Figure 4.36: Validation dice coefficient for Vanilla U-net (two-layer variant)

Correspondingly, figure 4.37 for the three-layer variant Vanilla U-net illustrates a similar result. The difference between the training and validation dice coefficient is typically small, which indicates that the model is generalizing well and is not overfitting to the training data; this is a good sign for the model's performance.

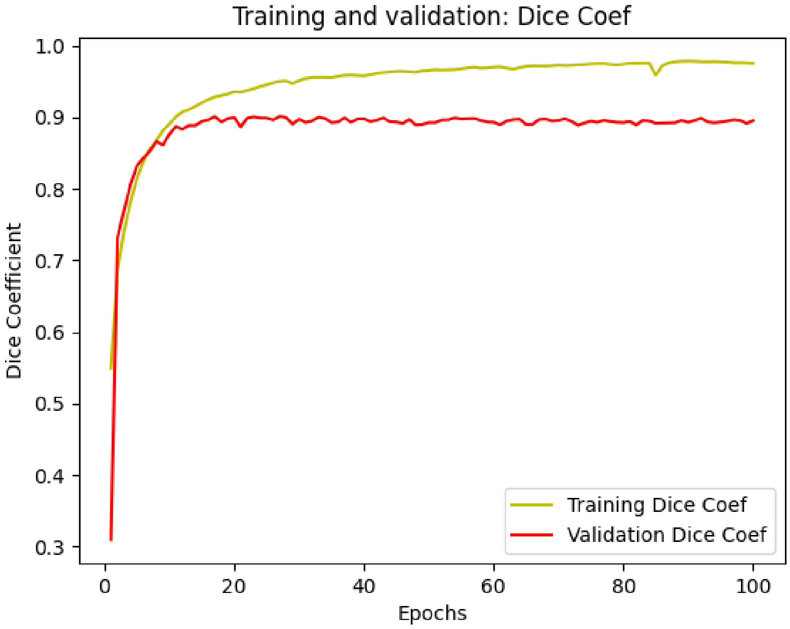


Figure 4.37: Validation dice coefficient for Vanilla U-net (three-layer variant)

Attention U-net In figure 4.38, we can see that the training dice coefficient increases as the model is trained and reaches a high value for the Attention U-net model with two convolutional layers. The validation dice coefficient also increases as the model is trained but plateaus at a lower value than the training dice coefficient and stays relatively stable. This suggests that the model is not overfitting to the training data.

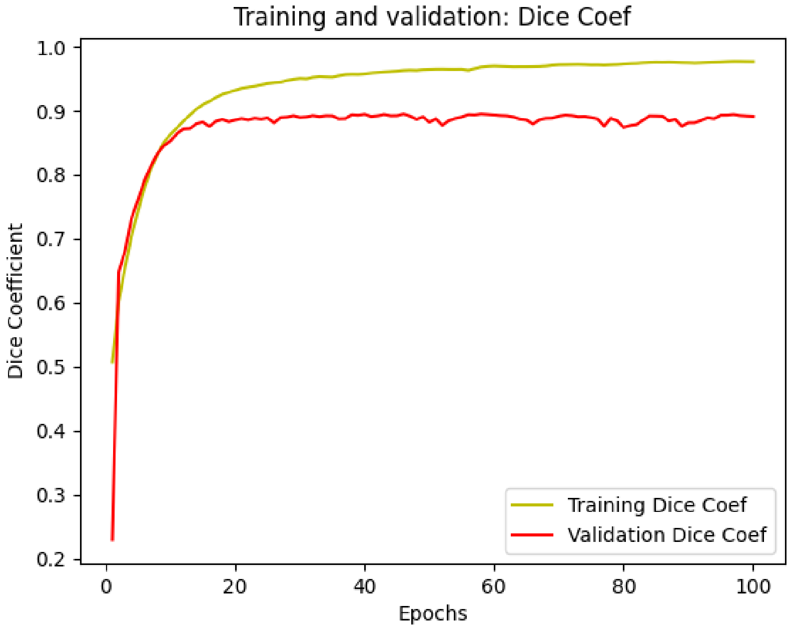


Figure 4.38: Validation dice coefficient for Attention U-net (two-layer variant)

Likewise, looking at figure 4.39, we can draw a similar conclusion. However, with the addition of one convolutional layer per block, the gap between the training dice coefficient and validation dice coefficient is narrower compared to the previous architecture.

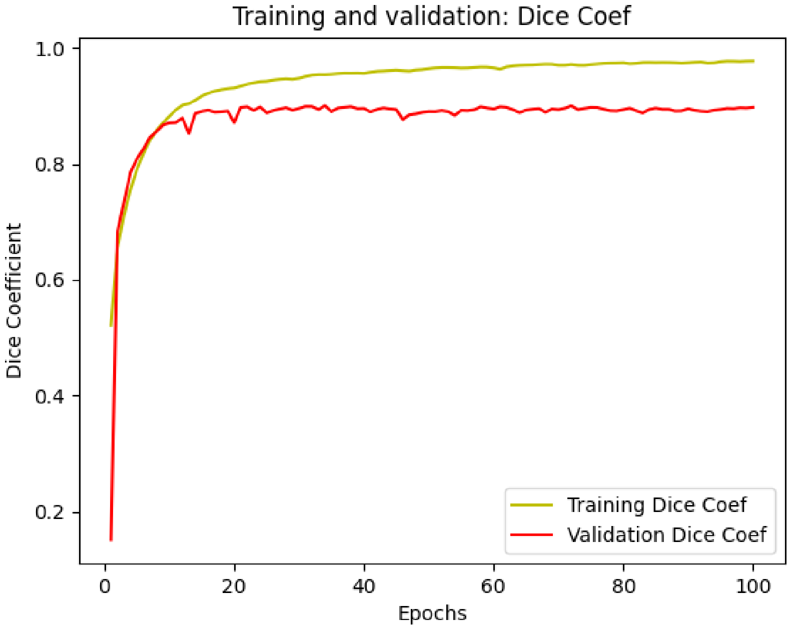


Figure 4.39: Validation dice coefficient for Attention U-net (three-layer variant)

Dense U-net As seen in figure 4.40, the performance of the model, Dense U-net (two-layer variant), seems decent as the validation dice coefficient is close to the training dice coefficient. This could be a sign that the model is able to learn the general pattern of image segmentation from the training data and applies the same to the validation data as well.

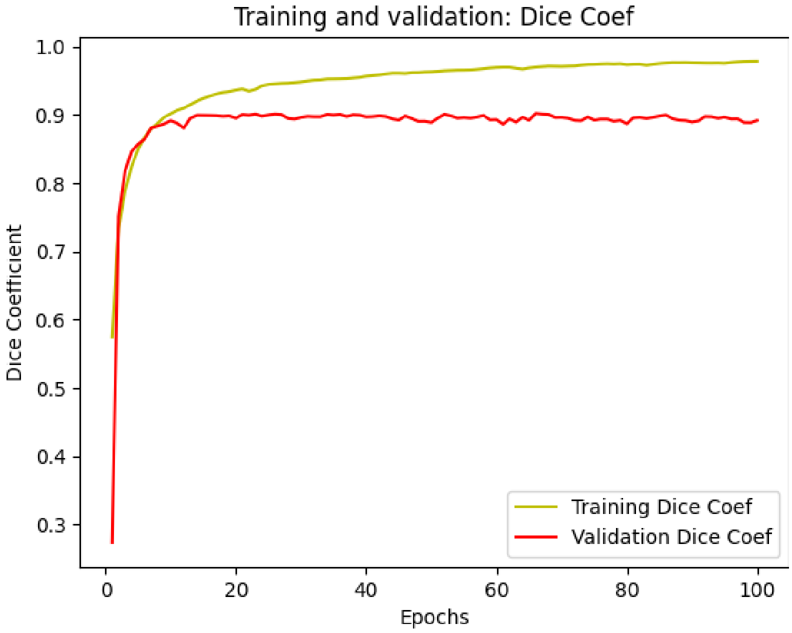


Figure 4.40: Validation dice coefficient for Dense U-net (two-layer variant)

Similarly, figure 4.41, depicts similar results for this model with the addition of one convolutional layer per block. The difference between the training and validation dice coefficient is typically small, indicating that the model is generalizing well and is not overfitting to the training data.

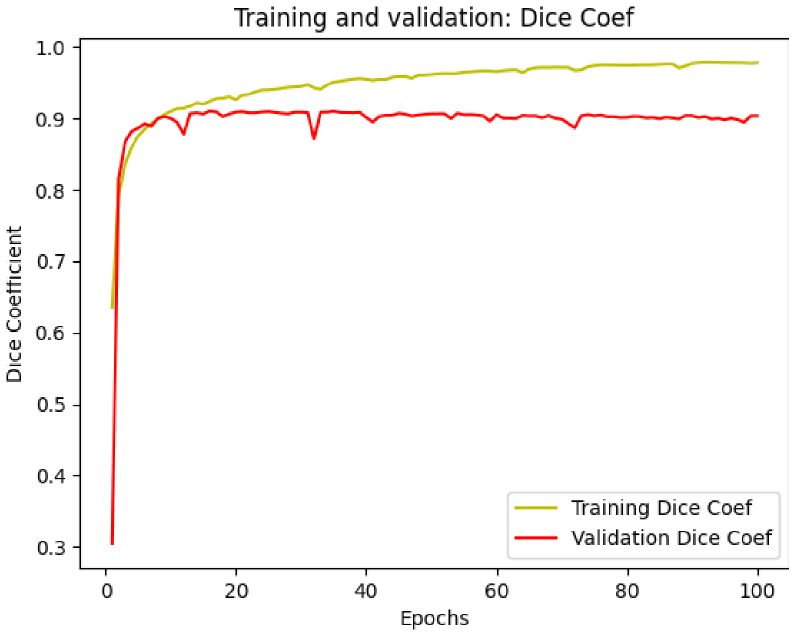


Figure 4.41: Validation dice coefficient for Dense U-net (three-layer variant)

R2 U-net We can see from figure 4.42 for R2 U-net (two-layer variant) that the validation dice coefficient is stable and close to the training dice coefficient. This is a positive indication of the model's performance on unseen data.

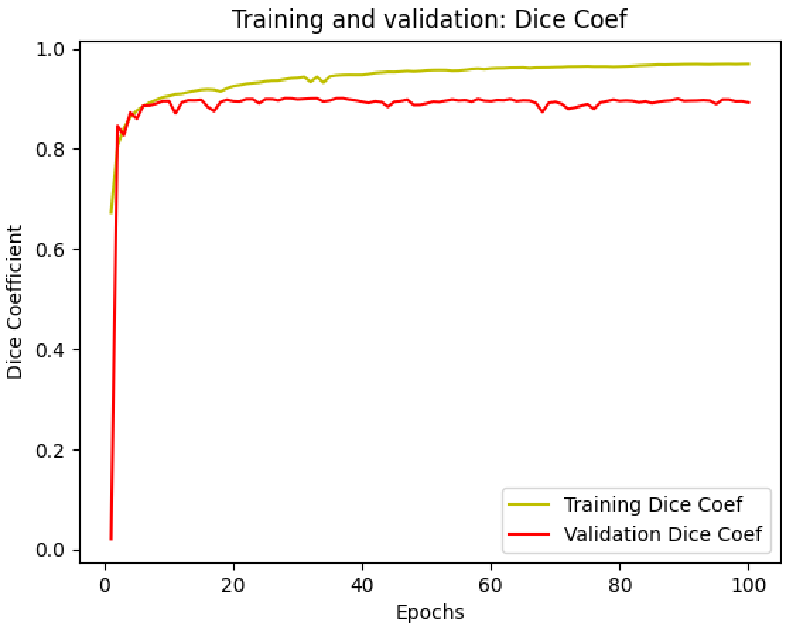


Figure 4.42: Validation dice coefficient for R2 U-net (two-layer variant)

As seen in figure 4.43 the validation dice coefficient appears to be fluctuating a bit in the early stages of training which is typical. However, the validation dice coefficient is stable after a certain point which suggests the model is not overfitting. Even the gap between the training dice coefficient and validation dice coefficient seems to be even narrower than the two-layer variant of this model.

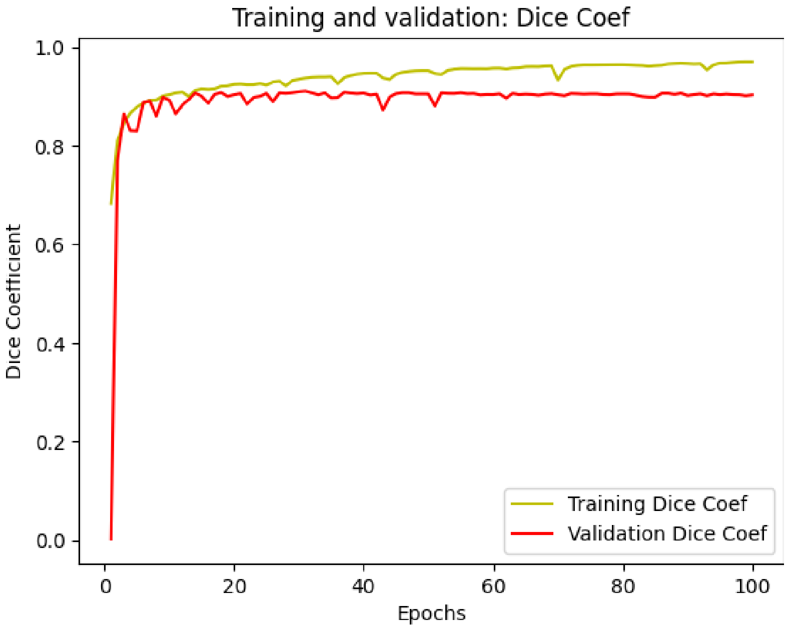


Figure 4.43: Validation dice coefficient for R2 U-net (three-layer variant)

Residual U-net Based on figure 4.44, it appears that Residual U-net (two-layer variant), is well-performing in terms of accuracy and generalization. The model's performance seems to be robust as both the training and validation dice coefficient have reached a high value, and also the validation dice coefficient is stable after a certain point which suggest model is not overfitting.

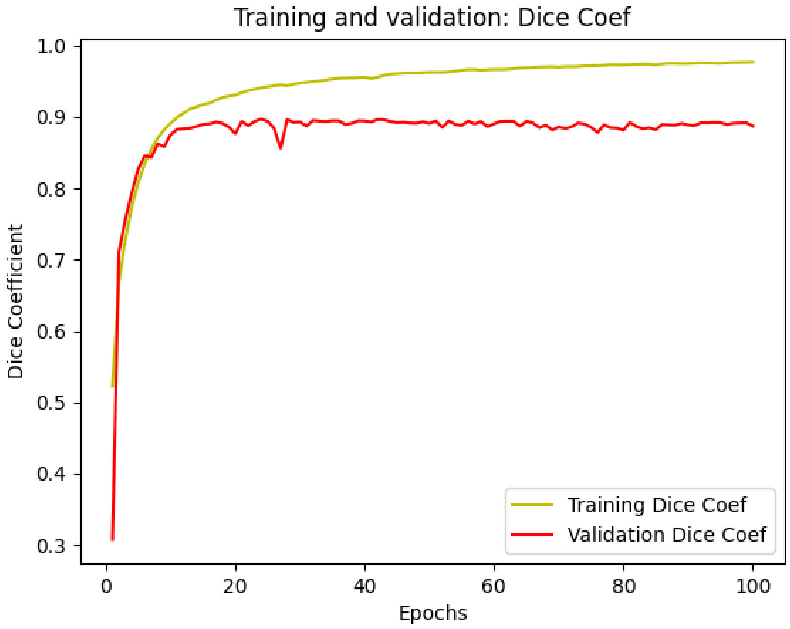


Figure 4.44: Validation dice coefficient for Residual U-net (two-layer variant)

Similarly, figure 4.45 illustrates that the gap between the training and validation dice coefficient is low and even narrower than the model previously mentioned. It usually depicts that this model, Residual U-net (with three convolutional layers), is not only fitting the training data well but also able to generalize to unseen data. This is a good indication that the model will perform well on new, unseen data.

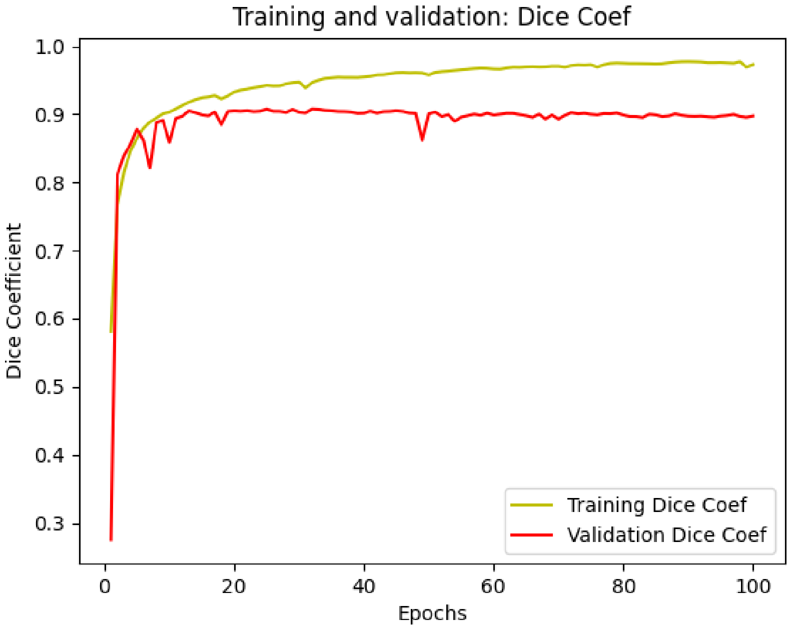


Figure 4.45: Validation dice coefficient for Residual U-net (three-layer variant)

SE U-net As seen in figure 4.46, the training dice coefficient and validation dice coefficient are relatively close together throughout the entire training process, with the validation dice coefficient staying only slightly behind the training dice coefficient. This suggests that the model, the two-layer variant SE U-net, is generalizing well and is not overfitting to the training data.

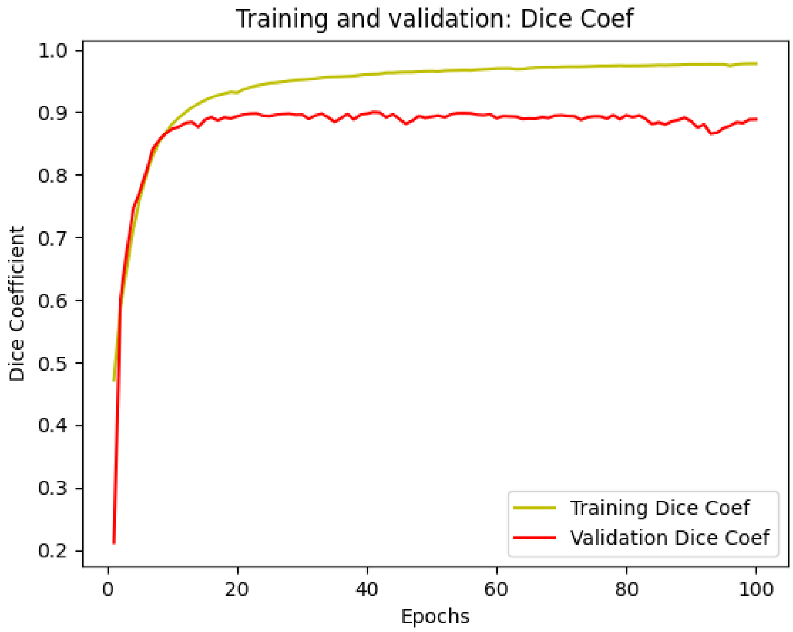


Figure 4.46: Validation dice coefficient for SE U-net (two-layer variant)

Likewise, figure 4.47 indicates that this model, with the addition of another convolution layer, is able to perform well on both the training and validation sets and is likely to have a good performance on unseen data. Based on this graph, it appears that the model is well-performing in terms of accuracy and generalization.

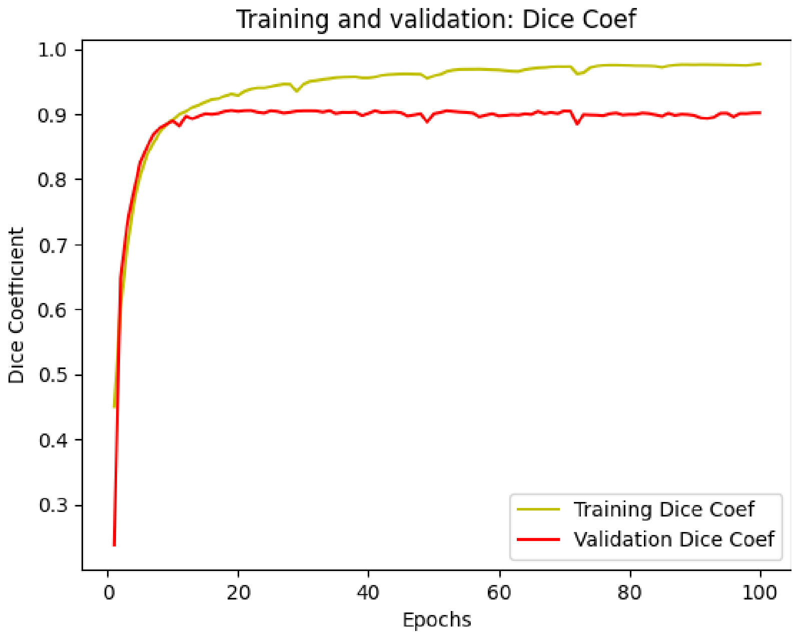


Figure 4.47: Validation dice coefficient for SE U-net (three-layer variant)

Intersection over Union (IoU)

The IoU is a valuable metric for assessing the effectiveness of a binary image segmentation model as it offers a straightforward, comprehensible way to evaluate the degree of overlap between predicted and ground truth segmentations.

As shown in figure 4.48, the intersection between the predicted foreground and the ground truth foreground is first found, and this value is then divided by the union of the predicted foreground and the ground truth foreground. This calculation is used to determine the IoU.

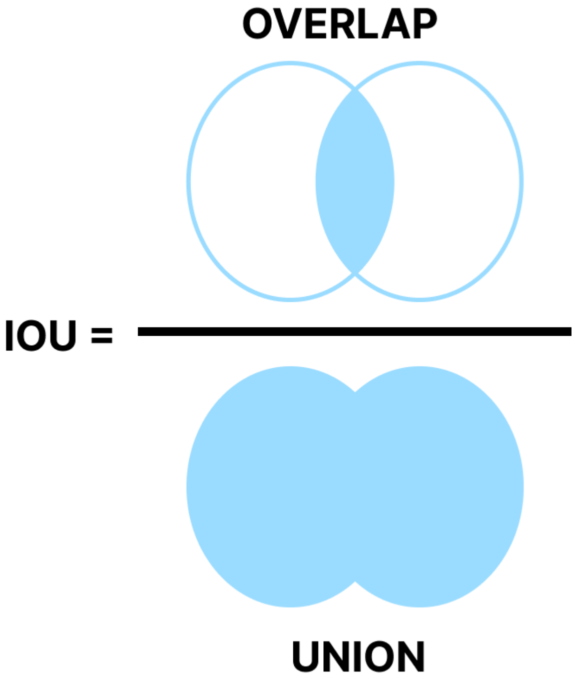


Figure 4.48: Intersection over Union (IoU) Venn diagram

The intersection over union (IOU) sets are produced by comparing the predicted segmentation set (A) with the ground truth segmentation set(B), and they are used to calculate the metrics true positive (TP), true negative (TN), false positive (FP), and false negative (FN).TP represents the number of pixels that the model has correctly identified as teeth (A ∩ B), TN represents the number of pixels that the model has correctly identified as background (A' ∩ B'), FP represents the number of pixels that the model has incorrectly identified as teeth (A ∩ B'), and FN represents the number of pixels that the model has incorrectly identified as background (A' ∩ B). A high number of TP and TN and a low number of FP and FN in the teeth segmentation are desirable signs that the model is correctly classifying the teeth and background.

The following equation 4.6 defines IoU:

$$\mathrm{IoU} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN}} \tag{4.6}$$

Vanilla U-net In figure 4.49, it can be seen that the train IoU and validation IoU are both increasing over time, and the difference between the two is small and stable for the Vanilla U-net (two-layer variant) architecture.

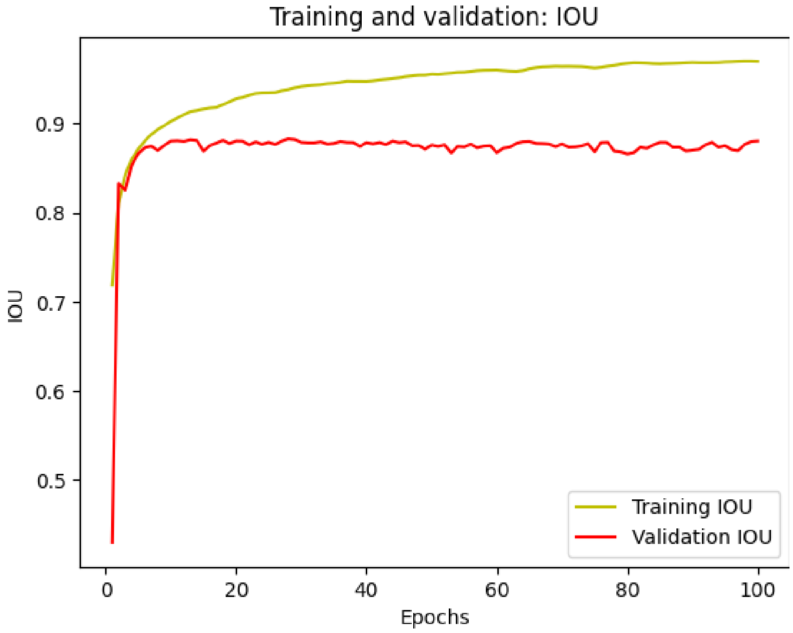


Figure 4.49: Validation IoU for Vanilla U-net (two-layer variant)

Correspondingly, figure 4.50 for the three-layer variant Vanilla U-net illustrates a similar result. The difference between the training and validation IoU is typically small, which indicates that the model is generalizing well and is not overfitting to the training data; this is a good sign for the model's performance.

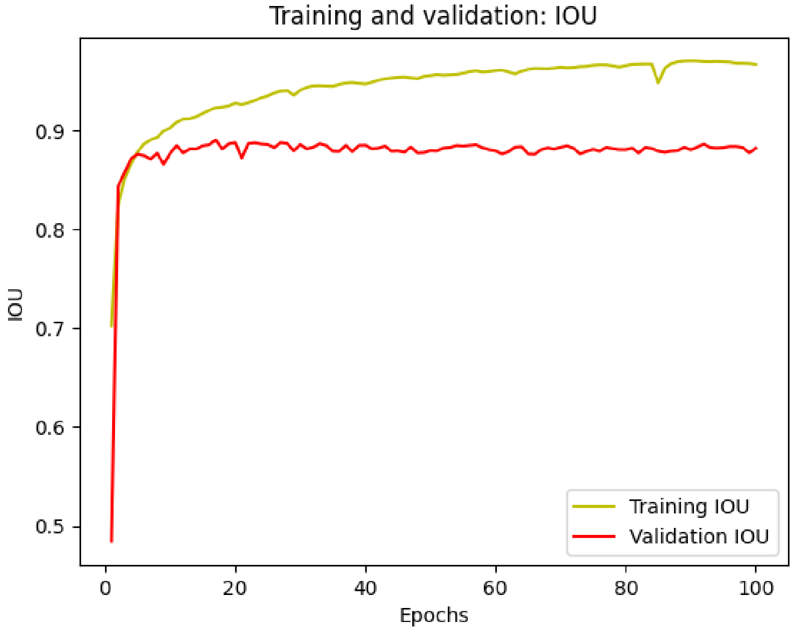


Figure 4.50: Validation IoU for Vanilla U-net (three-layer variant)

Attention U-net In figure 4.51, we can see that the training IoU increases as the model is trained and reaches a high value for the Attention U-net model with two convolutional layers. The validation IoU also increases as the model is trained but plateaus at a lower value than the training IoU and stays relatively stable. This suggests that the model is not overfitting to the training data.

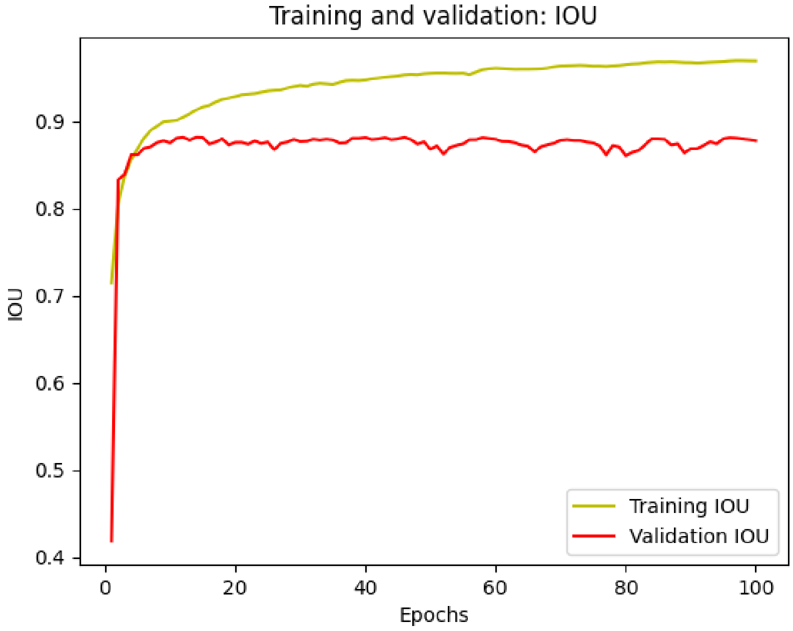


Figure 4.51: Validation IoU for Attention U-net (two-layer variant)

Likewise, looking at figure 4.52, we can draw a similar conclusion. However, with the addition of one convolutional layer per block, the gap between the training IoU and validation IoU is narrower compared to the previous architecture.

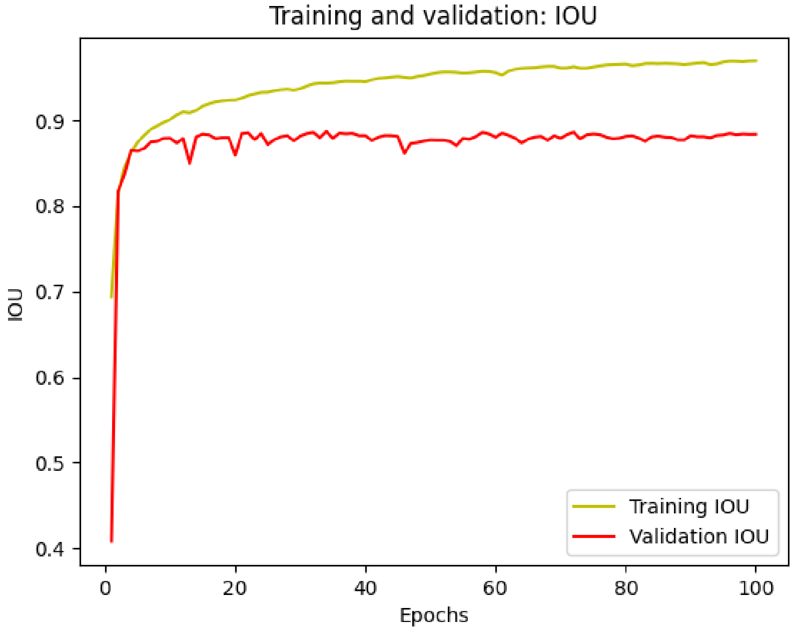


Figure 4.52: Validation IoU for Attention U-net (three-layer variant)

Dense U-net As seen in figure 4.53, the performance of the model, Dense U-net (two-layer variant), seems decent as the validation IoU is close to training IoU. This could be a sign that the model is able to learn the general pattern of image segmentation from the training data and applies the same to the validation data as well.

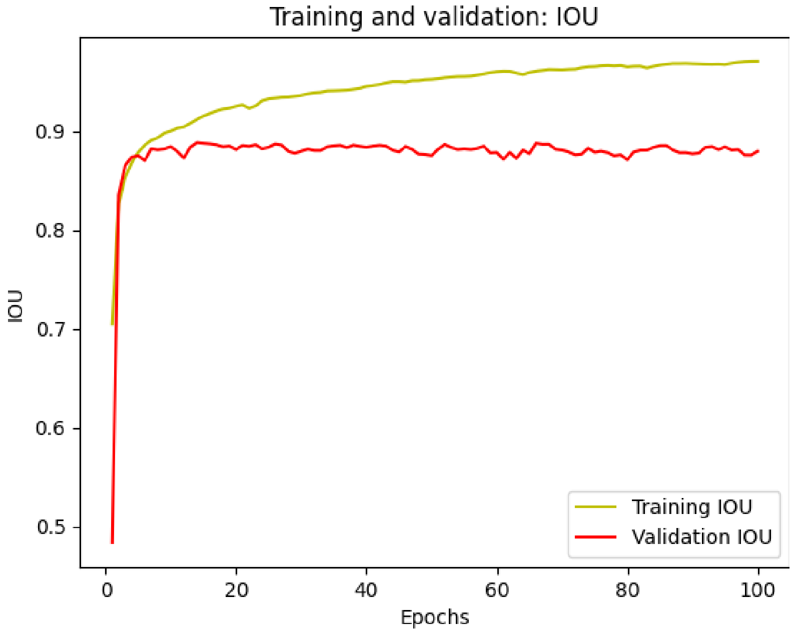


Figure 4.53: Validation IoU for Dense U-net (two-layer variant)

Similarly, figure 4.54, depicts similar results for this model with the addition of one convolutional layer per block. The difference between the training and validation IoU is typically small, indicating that the model is generalizing well and is not overfitting to the training data.

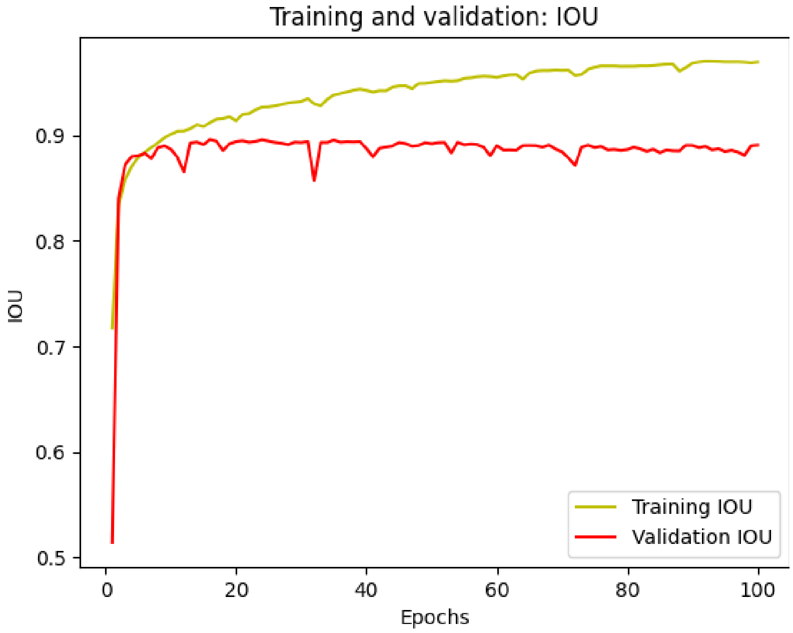


Figure 4.54: Validation IoU for Dense U-net (three-layer variant)

R2 U-net We can see from figure 4.55 for R2 U-net (two-layer variant) that the validation IoU is stable and close to training IoU. This is a positive indication of the model's performance on unseen data.

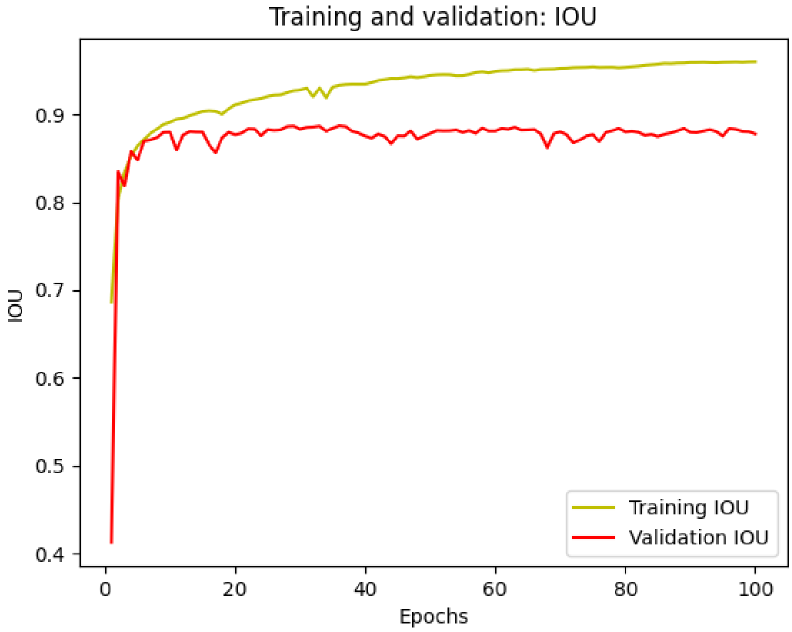


Figure 4.55: Validation IoU for R2 U-net (two-layer variant)

As seen in figure 4.56 for R2 U-net (three-layer variant), the validation IoU appears to be fluctuating a bit in the early stages of training, which is typical. However, the validation IoU is stable after a certain point which suggests the model is not overfitting. Even the gap between training IoU and validation IoU seems to be even narrower than the two-layer variant of this model.

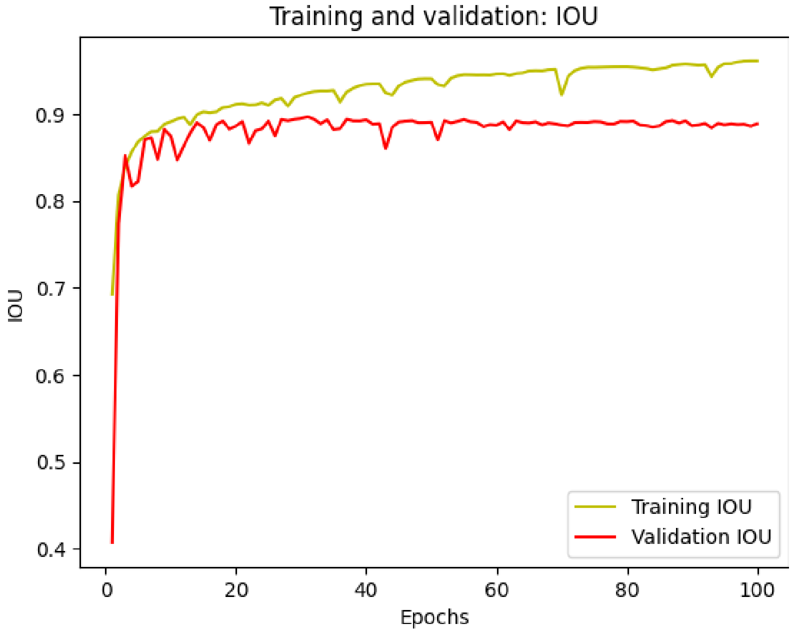


Figure 4.56: Validation IoU for R2 U-net (three-layer variant)

Residual U-net Based on figure 4.57, it appears that Residual U-net (two-layer variant) is well-performing in terms of accuracy and generalization. The model's performance seems to be robust as both training and validation IoU have reached a high value, and also, the validation IoU is stable after a certain point which suggests the model is not overfitting.

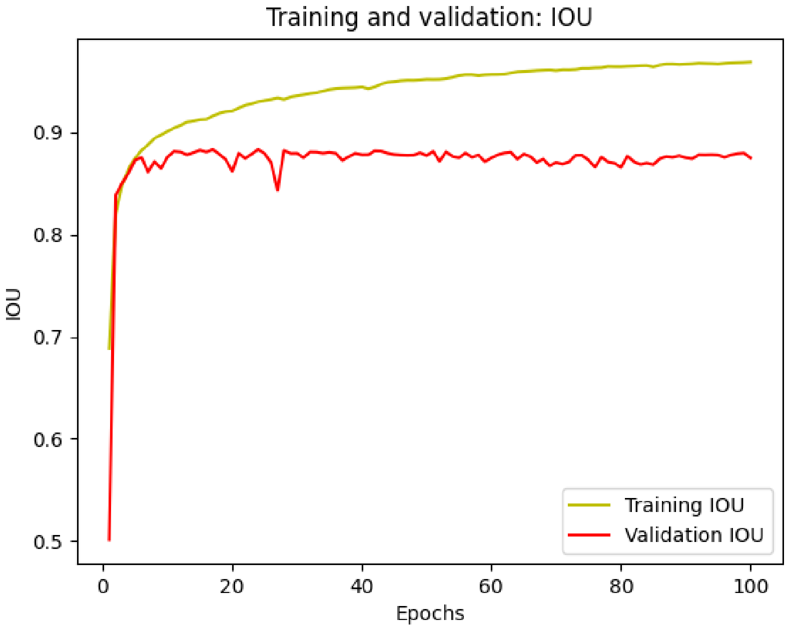


Figure 4.57: Validation IoU for Residual U-net (two-layer variant)

Similarly, figure 4.58 illustrates that the gap between the training and validation IoU is low and even narrower than the model previously mentioned. It usually depicts that this model, Residual U-net (with three convolutional layers), is not only fits the training data well but also able to generalize to unseen data. This is a good indication that the model will perform well on new, unseen data.

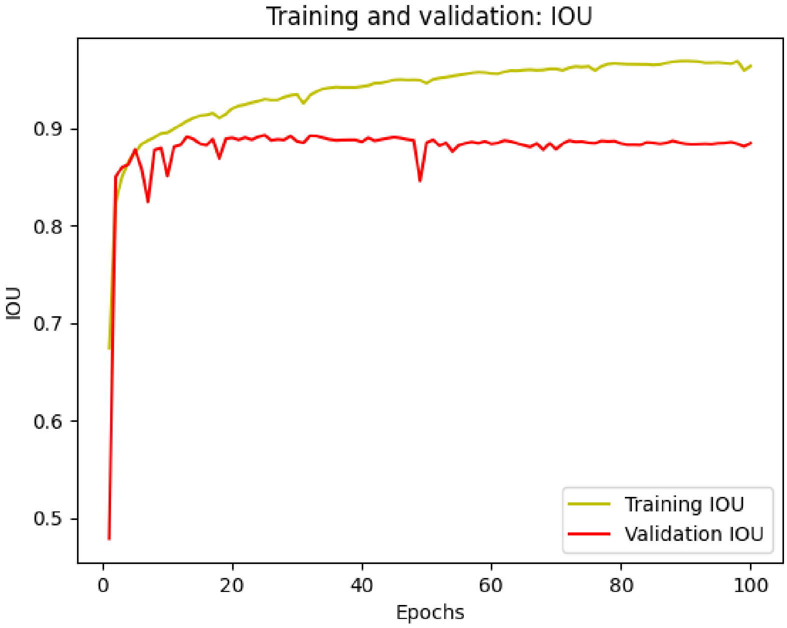


Figure 4.58: Validation IoU for Residual U-net (three-layer variant)

SE U-net As seen in figure 4.59, the training IoU and validation IoU are relatively close together throughout the entire training process, with the validation IoU staying only slightly behind the training IoU. This suggests that the model, the two-layer variant SE U-net, is generalizing well and is not overfitting to the training data.

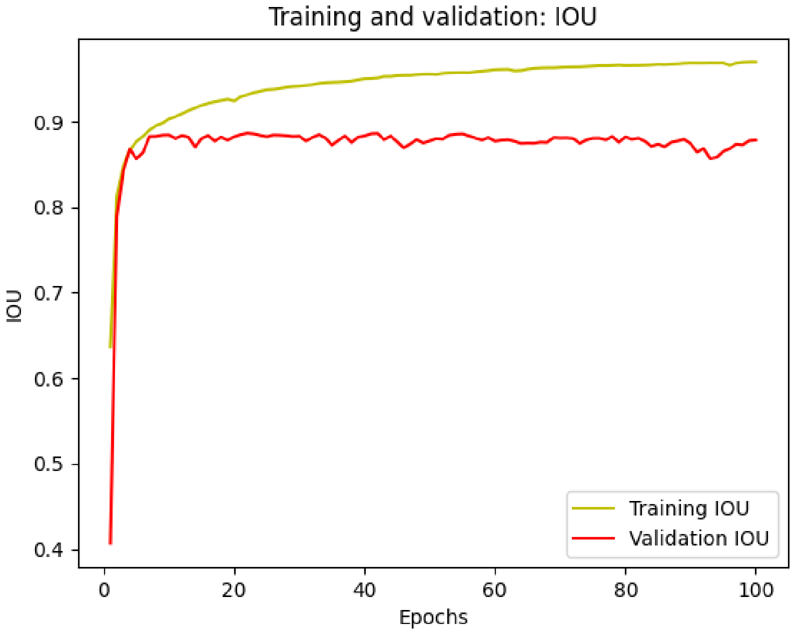


Figure 4.59: Validation IoU for SE U-net (two-layer variant)

Likewise, figure 4.60 indicates that this model, with the addition of another convolution layer, is able to perform well on both the training and validation sets and is likely to have a good performance on unseen data. Based on this graph, it appears that the model is well-performing in terms of accuracy and generalization.

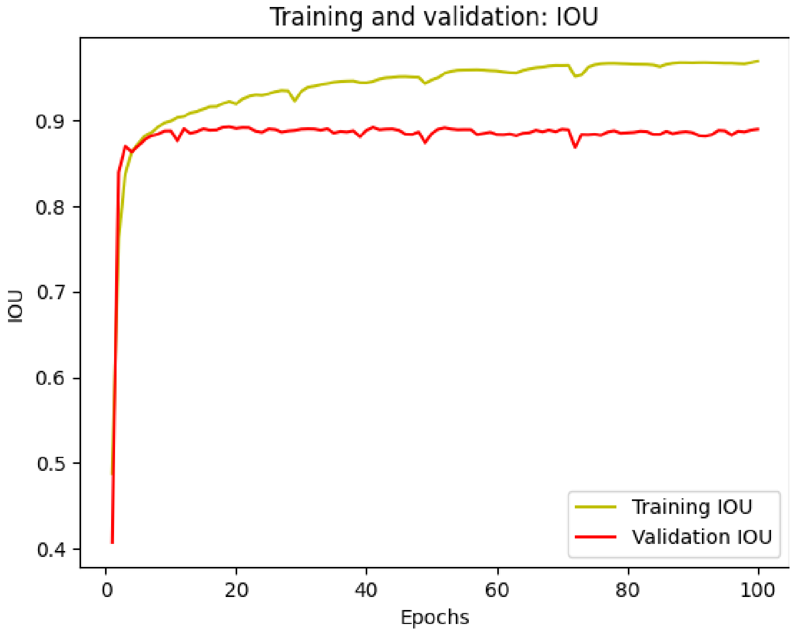


Figure 4.60: Validation IoU for SE U-net (three-layer variant)

4.4.2 Tabular Evaluation

We used a confusion matrix to summarize the effectiveness of all U-net variants because it is clear and easy to understand. For a certain test set, it generates a table that is used to compare the predicted class labels to the ground truth class labels and gives a summary of the number of accurate and inaccurate predictions the model made.

Predicting the class label for each pixel inside an image is the aim of image segmentation. The performance of a model may be assessed using a confusion matrix by evaluating the predicted class labels with the true class labels for a certain test set. It depicts the number of true positives, true negatives, false positives, and false negatives. The general format of a confusion matrix table for binary image segmentation is as follows:

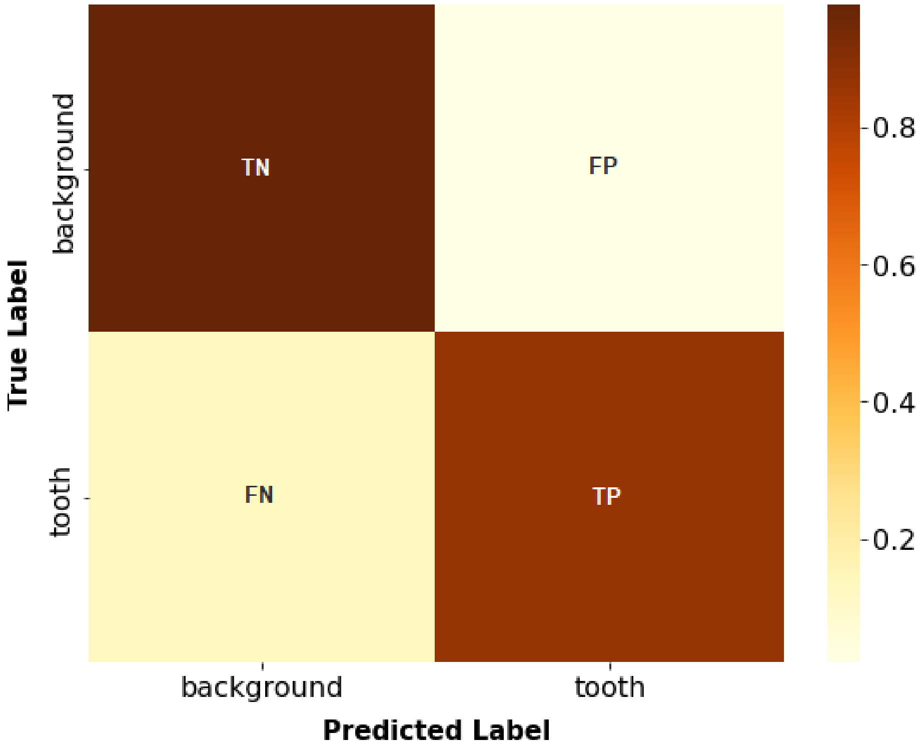


Figure 4.61: Basic 2 × 2 confusion matrix

In figure 4.61, the columns correspond to the predicted class labels, whereas the rows correspond to the true class labels (foreground and background) for a confusion matrix table. The cells of the table contain the following quantities:

True positives (TP): The portion of pixels that are correctly classified as foreground (teeth).
False negatives (FN): The portion of pixels that are incorrectly classified as background (i.e., they should be foreground).
False positives (FP): The portion of pixels that are incorrectly classified as foreground (i.e., they should be background).
True negatives (TN): The portion of pixels that are correctly classified as background.

Vanilla U-net As illustrated in figure 4.62, the confusion matrix table for this two-layer variant model is quite decent. 87% pixels of teeth that were correctly identified by the model as teeth. Additionally, the true negative rate is also very high at 0.98, meaning that 98% of the pixels of the background were correctly identified as background. The false positive rate is relatively low at 0.02, indicating that only 2% of pixels of the background were incorrectly identified as teeth. Similarly, 13% of pixels of teeth were incorrectly identified as background. Overall, this model seems to be performing well in terms of accurately identifying teeth and background.

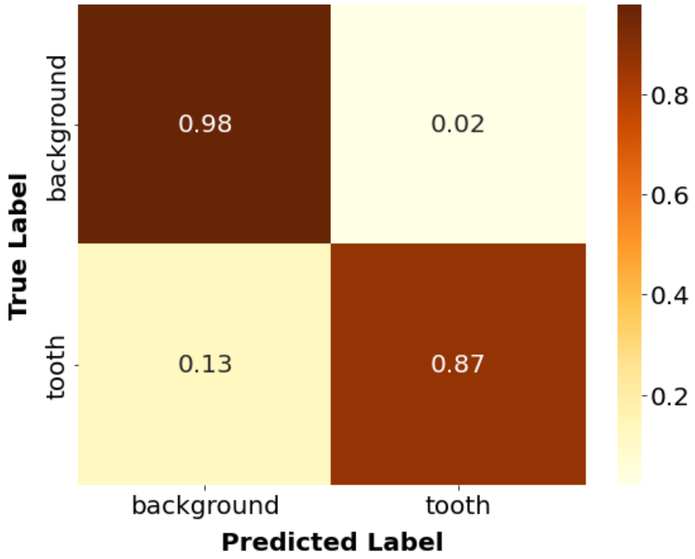


Figure 4.62: Confusion Matrix for Vanilla U-net (two-layer variant)

Figure 4.63 depicts that the three-layer variant shows an improvement in the false negative rate compared to the previous variant, indicating that the model is better at identifying teeth pixels. Overall, this model has a high accuracy in identifying teeth pixels and is an improvement over the first model.

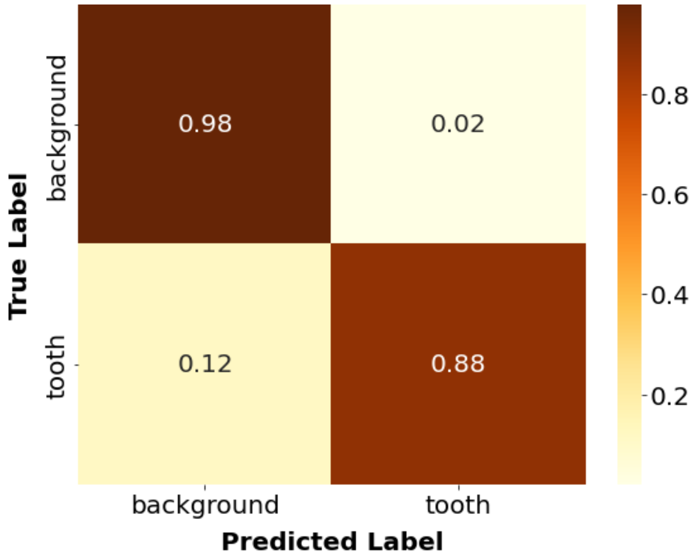


Figure 4.63: Confusion Matrix for Vanilla U-net (three-layer variant)

Attention U-net According to figure 4.64 Attention U-net with two convolutional layers has outperformed Vanilla U-net (by $2-3\%$) in all four TP, FP, TN, and FN quantities on the test sets. This indicates that this model has a high overall accuracy where the model is able to correctly classify a majority of the images as either teeth or background. Additionally, the high true positive rate of 0.9 and the low false negative rate of 0.1 are, in fact, the highest among the two-layer variants.

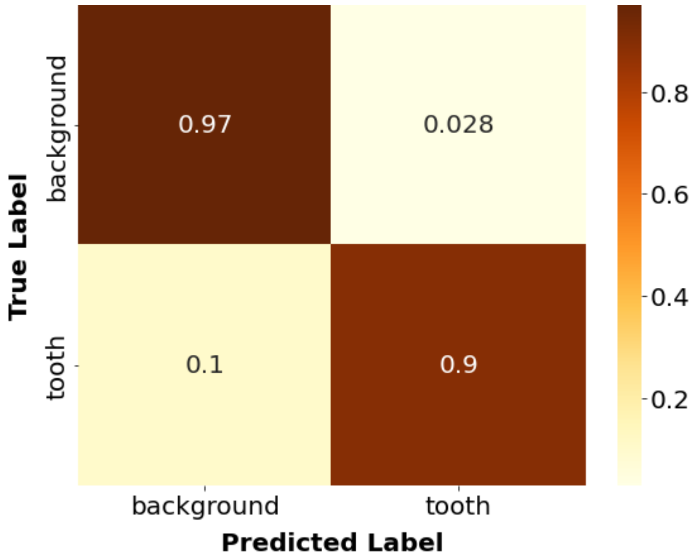


Figure 4.64: Confusion Matrix for Attention U-net (two-layer variant)

Figure 4.65 demonstrates that the confusion matrix for the three-layer variant is similar to the first one. However, the true negative rate (0.98) and false positive rate (0.024) are even greater and smaller than the previous variant. This indicates that the model has improved misclassifying background as teeth. Both models have similar performance, but the second one might be slightly more accurate as per its confusion matrix on the test sets.

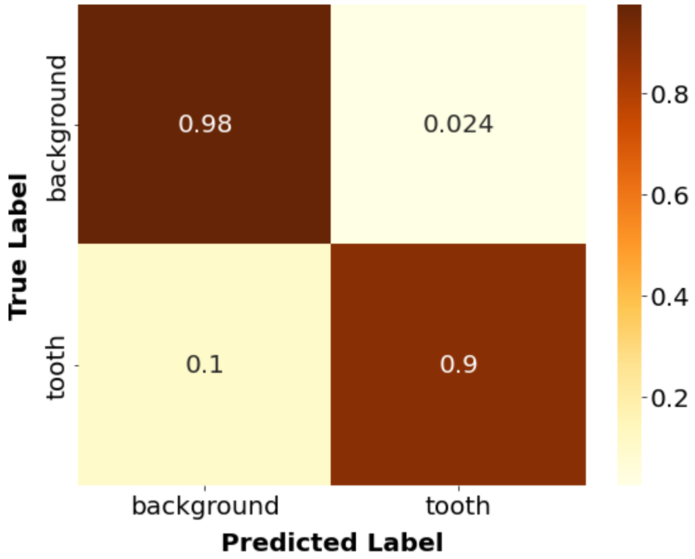


Figure 4.65: Confusion Matrix for Attention U-net (three-layer variant)

Dense U-net As shown in figure 4.66, Dense U-net (two convolution layers) has a high true positive rate of 0.87 which means the model is able to accurately identify teeth in the images. Additionally, the true negative rate is also high at 0.98, which means that the model is also able to correctly identify the background in the test sets. This combination of high true positive and true negative rates is a positive indication that the model is performing well overall.

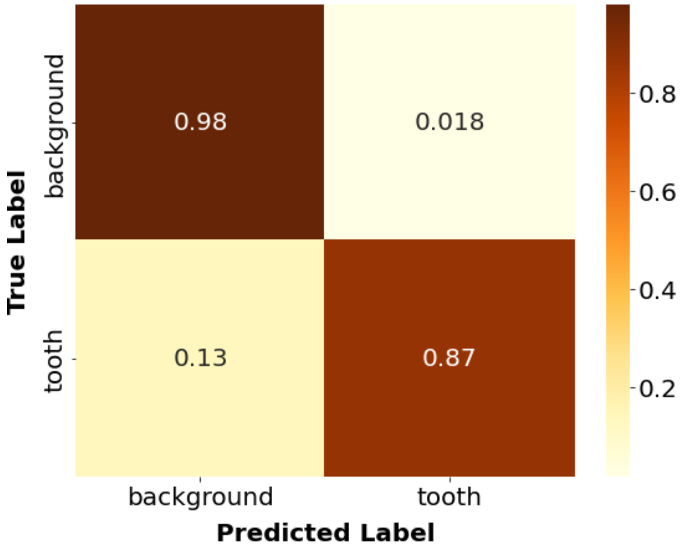


Figure 4.66: Confusion Matrix for Dense U-net (two-layer variant)

Figure 4.67 depicts this three-layer variant has an even higher true positive rate of 0.89, which is a great indication that the model is accurately identifying teeth in the test images. Additionally, the false negative rate is also lower at 0.11, which means that the model is identifying a larger proportion of teeth in the images. Furthermore, the true negative rate is also high at 0.98, which confirms that the model is also able to correctly identify the background in the images. Overall, the second model is performing slightly better than the first one, thanks to the higher true positive rate and lower false negative rate.

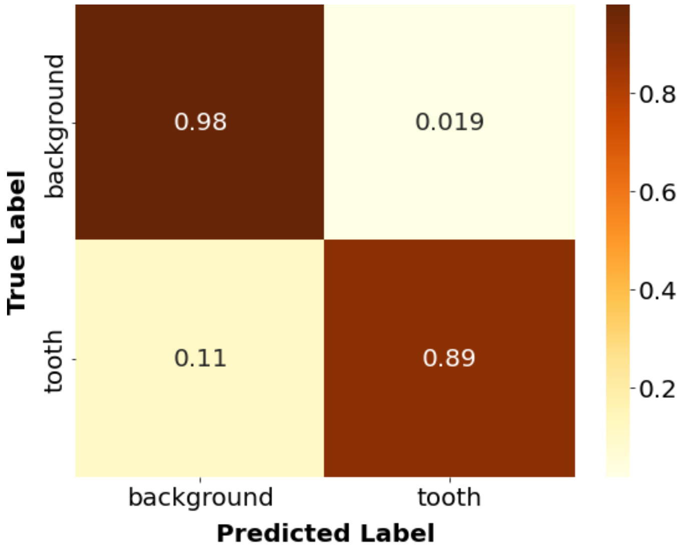


Figure 4.67: Confusion Matrix for Dense U-net (three-layer variant)

R2 U-net Figure 4.68 displays that R2 U-net (two-layer variant) has an outstanding true negative rate of 0.98, which means that the model is very accurate at identifying non-teeth. Additionally, the false positive rate of 0.025 is very low, indicating that the model is not frequently misclassifying non-teeth as teeth. Furthermore, the true positive rate of 0.89 is also very high, which means that the model is accurately identifying teeth in the images.

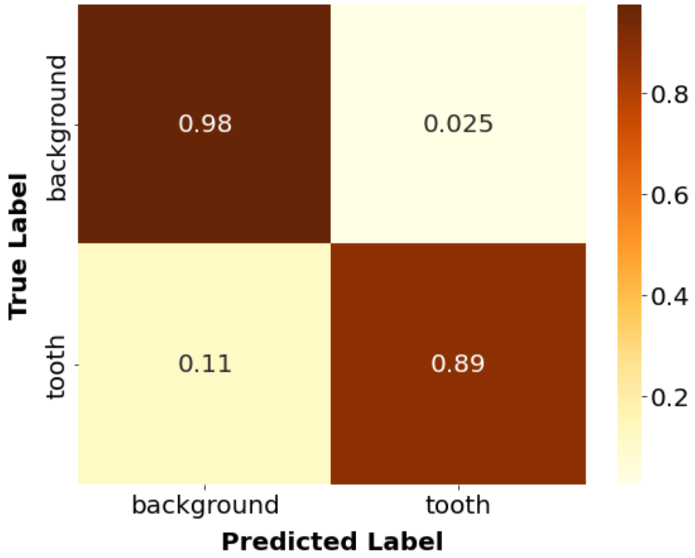


Figure 4.68: Confusion Matrix for R2 U-net (two-layer variant)

Figure 4.69 illustrates that R2 U-net (three-layer variant) is also very similar to the previous variant while classifying the background pixels, with a true negative rate of 0.98 and a very low false positive rate of 0.024. However, the true positive rate of 0.91 is higher than the two-layer variant, which means that the model is accurately identifying teeth even better with very high precision. Additionally, with an FN rate of 0.087, it is even more accurate than the first model. This variant of R2 U-net is the best among all variants of all architectures.

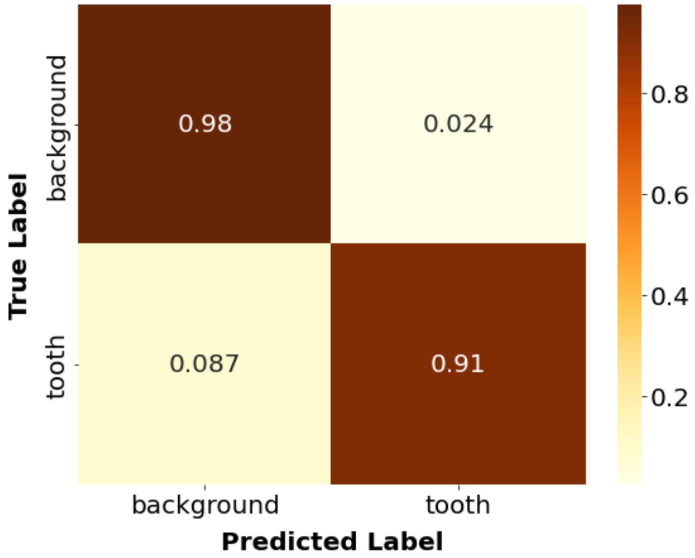


Figure 4.69: Confusion Matrix for R2 U-net (three-layer variant)

Residual U-net As displayed in figure 4.70, Residual U-net (two-layer variant) has a very high true negative rate of 0.98, indicating that the model is very good at correctly identifying non-teeth. Additionally, the false positive rate of 0.019 is very low, showing that the model is not frequently misclassifying background as teeth. Although the false negative rate is at 0.14 and the true positive rate is at 0.86, the model still performs well overall. This is because even though the model might miss some teeth in the images, it is still correctly identifying the majority of them.

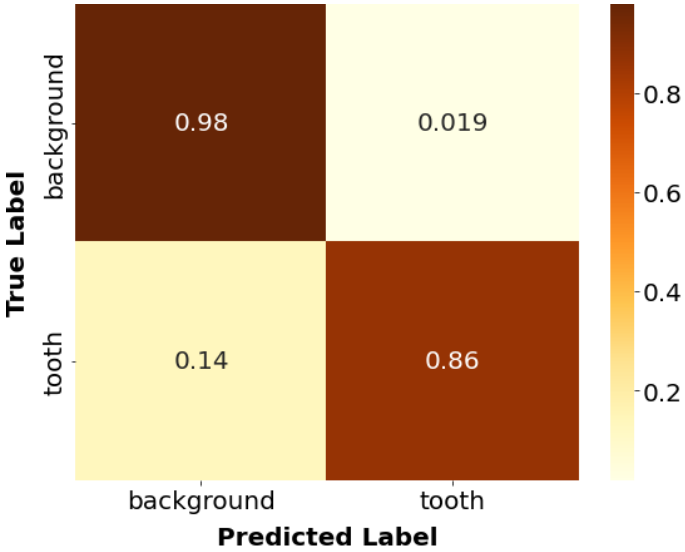


Figure 4.70: Confusion Matrix for Residual U-net (two-layer variant)

Figure 4.71 demonstrates that this three-layer variant of the model has an even better performance than the first one. With a true negative rate of 0.98, a false positive rate of 0.018, and a true positive rate of 0.88, the model is able to correctly identify almost all instances of teeth and non-teeth in the test sets. The false negative rate of 0.12 is lower than the first one, indicating that the model is better at identifying all instances of teeth. With the results of these two models, it's obvious that the second one is more accurate and reliable.

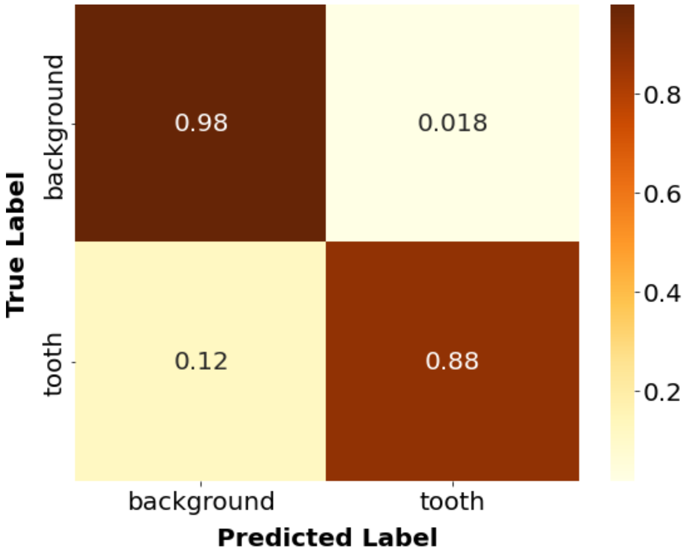


Figure 4.71: Confusion Matrix for Residual U-net (three-layer variant)

SE U-net As illustrated in figure 4.72, SE U-net's performance on the test sets is identical to both of Residual U-net's variants. However, this model's false positive rate of 0.016 is relatively lower. Additionally, it has a high true negative rate of 0.98, indicating that the model is very effective at correctly identifying non-teeth in the images. The False Negative rate of 0.14 is relatively low, meaning the model can correctly identify most of the teeth in the images.

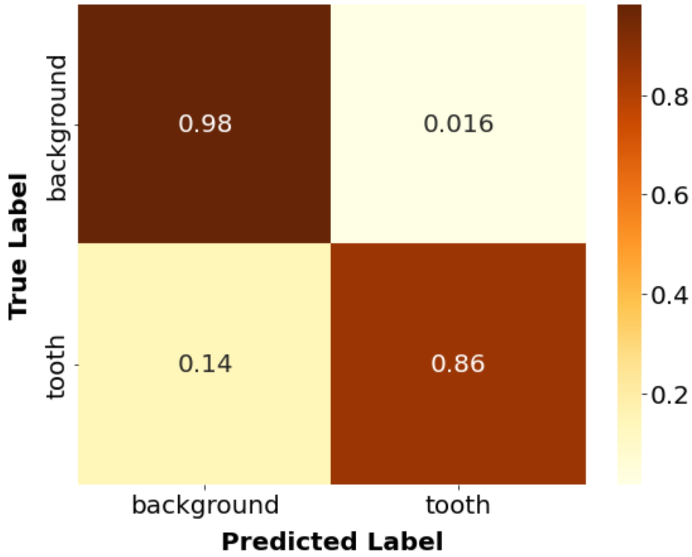


Figure 4.72: Confusion Matrix for SE U-net (two-layer variant)

Figure 4.73 depicts this model's three-layer variant also has a high true negative rate of 0.98, similar to the first variant. The False Negative rate of 0.12 is even lower than the previous variant, meaning that the model is even better at correctly identifying teeth in the images. Additionally, the false positive rate of 0.019 is also low, indicating that the model is not frequently misclassifying non-teeth as teeth. Both models have similar accuracy, but this variant is slightly better.

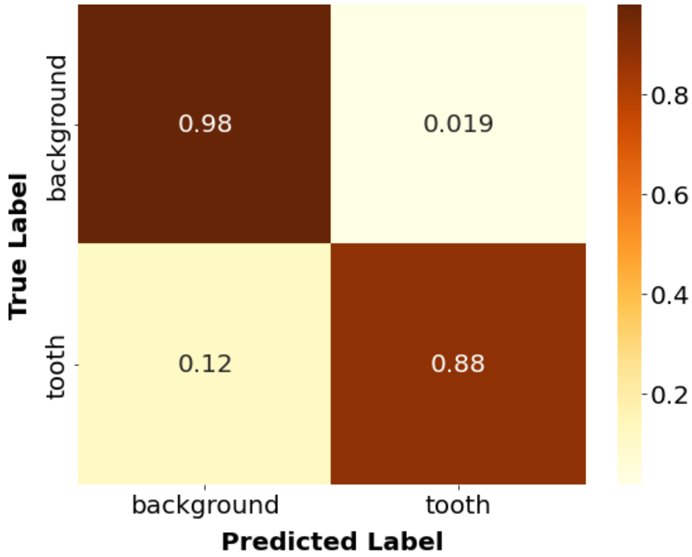


Figure 4.73: Confusion Matrix for SE U-net (three-layer variant)

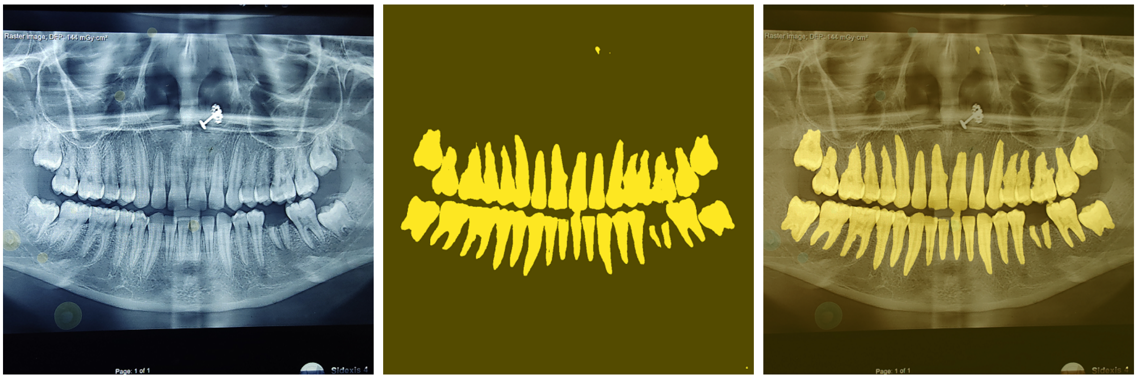
Chapter 5

Result Analysis

The segmentation performance of the OPG teeth image dataset was examined using six distinct U-Net variations in this study. Using two and three convolutional blocks per layer, comparisons of each design were also made.

5.1 Visual Inspection

Visual inspection is a valuable technique for analyzing the results of different segmentation models. It involves looking at the predicted segmentations and comparing them to the ground truth segmentations visually in order to get a sense of how well the models are performing.



(a) Sample image (b) Prediction on the image (c) Overlay of prediction

Figure 5.1: Sample of Visual Inspection

Figure 5.1 shows how it is done by overlaying the predicted segmentation on top of the original image. Later, we can analyze the results produced by different segmentation models. By comparing the overlaid images for different models, we can get a sense of which model is doing the best job of accurately predicting the class labels for each pixel in the image.

Vanilla U-net As depicted in figure 5.2, the segmented image predicted by the Vanilla U-net (two convolutional layers) shows a clear separation of the teeth from the surrounding tissue, with minimal error in the segmentation. The model has accurately captured the shape and size of the teeth for most of the parts.

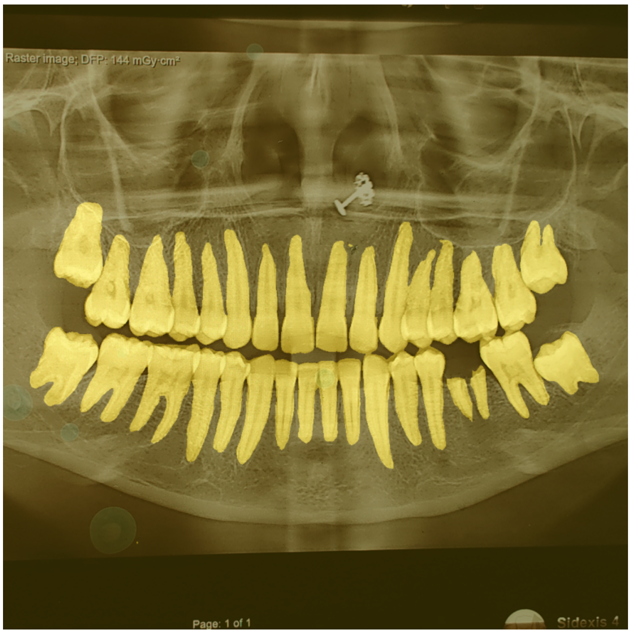


Figure 5.2: Visual Inspection for Vanilla U-net (two-layer variant)

In figure 5.3, the segmentation quality of the incisors, canines, and molars was enhanced by this model with an additional convolutional layer, although a few pixels of the gums around the premolar teeth were misclassified.

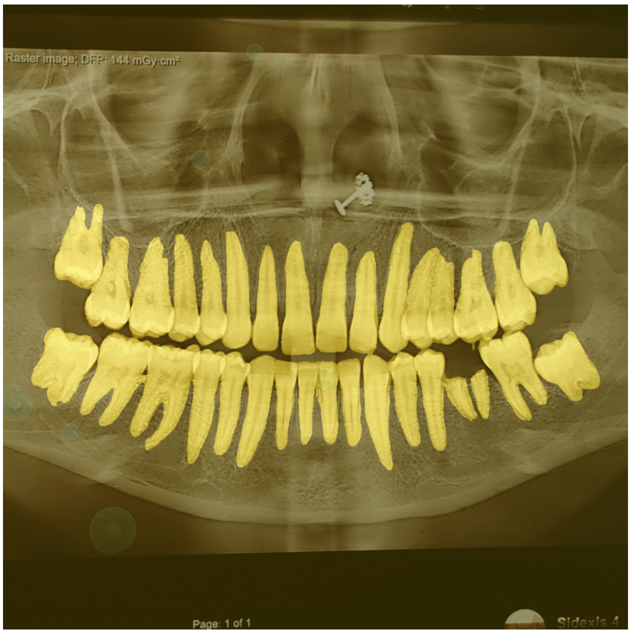


Figure 5.3: Visual Inspection for Vanilla U-net (three-layer variant)

Attention U-net As seen from figure 5.4, Attention U-net (two-layer variant) struggled quite a bit in accurately identifying the teeth in the test image. Even though most of the teeth were correctly segmented, there were a few occasions where a few pixels were misclassified, such as jawbone, in between teeth and gums.

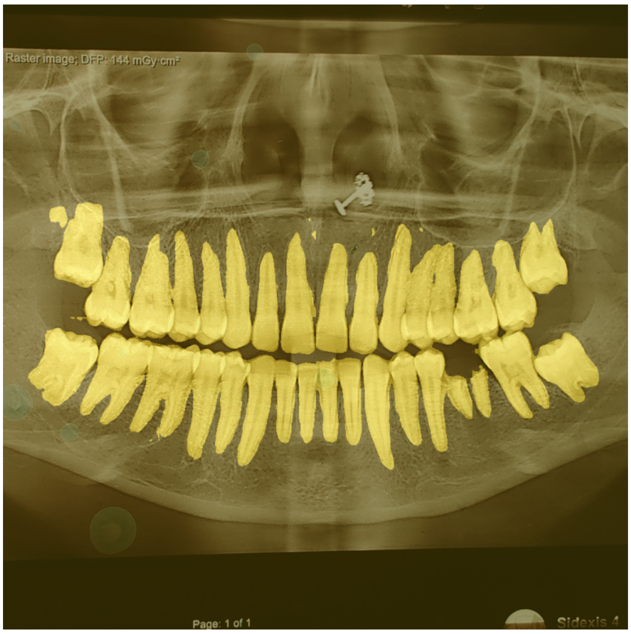


Figure 5.4: Visual Inspection for Attention U-net (two-layer variant)

From figure 5.5, we can see that this model with an additional convolutional layer improved its performance in terms of accurately identifying the teeth in the test image. The overall segmentation quality was high.

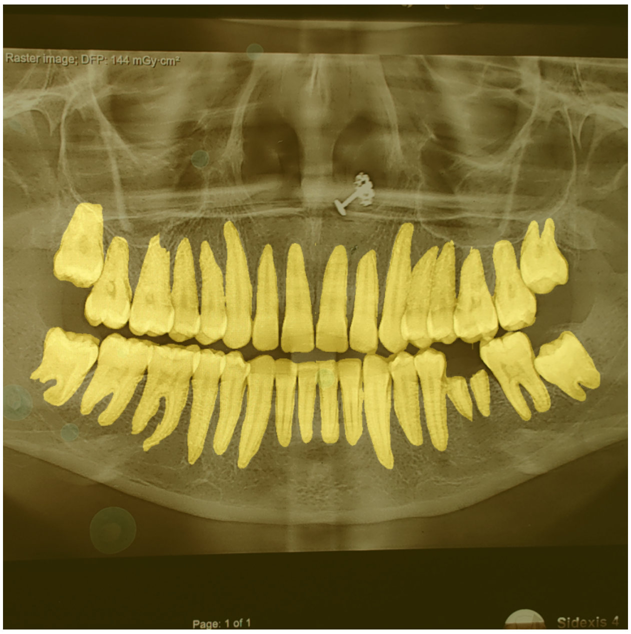


Figure 5.5: Visual Inspection for Attention U-net (three-layer variant)

Dense U-net Inspecting figure 5.6, we can see that Dense U-net (with two convolutional layers) has accurately segmented the teeth, with clear boundaries separating the teeth from the surrounding tissue. The segmentation results match well with the original image.

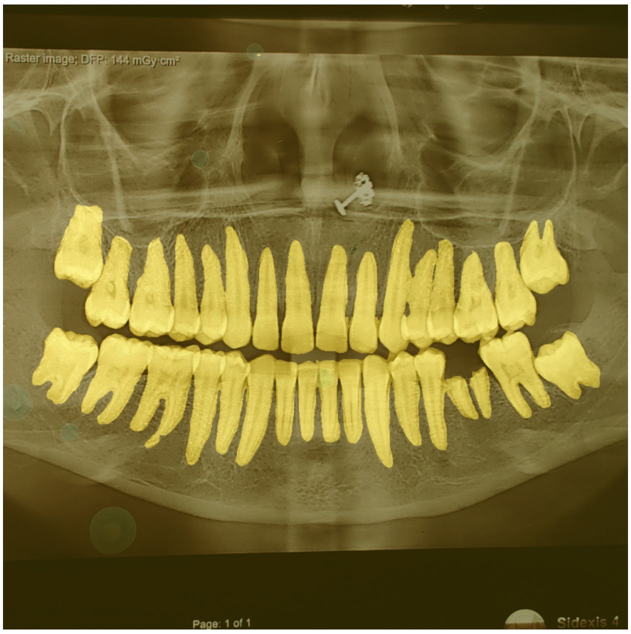


Figure 5.6: Visual Inspection for Dense U-net (two-layer variant)

figure 5.7 for Dense U-net (three-layer variant) shows a similar result. Except for the misclassified overlap of pixels between premolars on the upper-right, most of the edges for the teeth were even better classified by this model compared to its model with two convolutional layers.

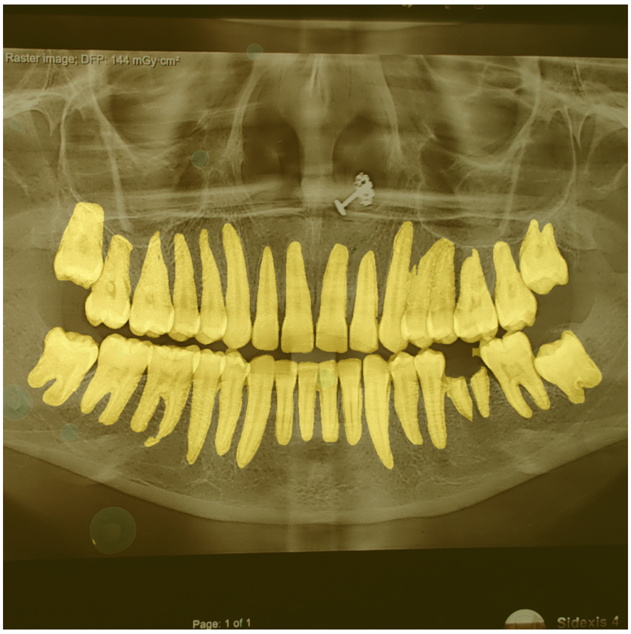


Figure 5.7: Visual Inspection for Dense U-net (three-layer variant)

R2 U-net As depicted in figure 5.8, we can see a couple of small segments of gums were misclassified by the model, R2 U-net, with convolutional layers per block. Excluding these minor errors, this model has correctly separated the teeth from the surrounding tissue, and the segmented image clearly shows the individual teeth.

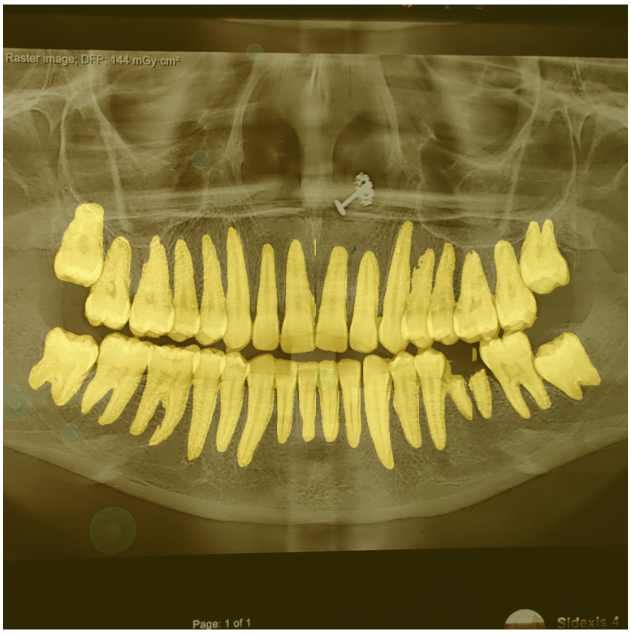


Figure 5.8: Visual Inspection for R2 U-net (two-layer variant)

As seen from figure 5.9, this model with an additional convolutional layer resolved its previous issue. Overall, the segmented image shows a clear separation of the teeth from the surrounding tissue, with minimal error in the segmentation.

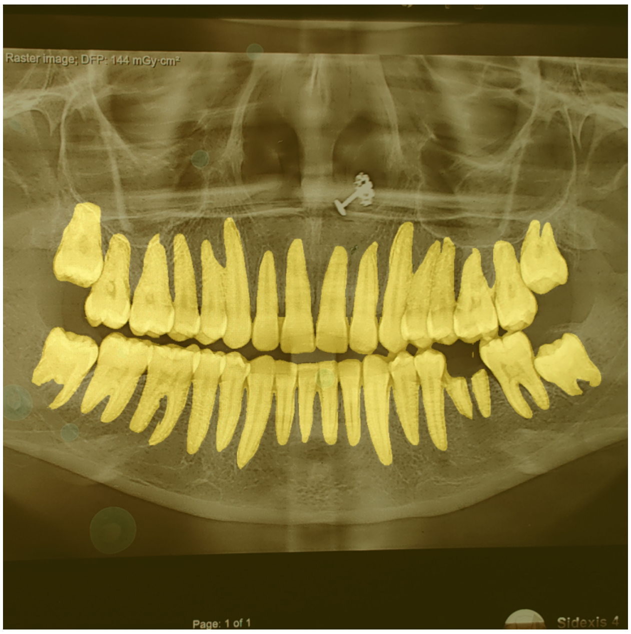


Figure 5.9: Visual Inspection for R2 U-net (three-layer variant)

Residual U-net Inspecting figure 5.10, it is clear that Residual U-net (two-layer variant) has accurately segmented the majority of the teeth in the images. The incisors, canines, and premolars are clearly defined and separated from the surrounding tissue. However, the model is struggling to define the edges of the wisdom teeth a bit.

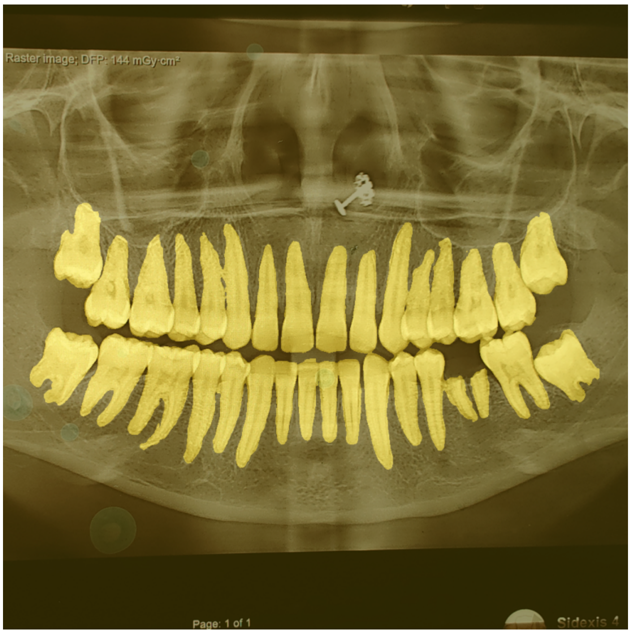


Figure 5.10: Visual Inspection for Residual U-net (two-layer variant)

From figure 5.11, we can see that additional convolutional layers per block helped this model to further improve its performance in terms of accurately identifying all the teeth, including even the edges of wisdom teeth.

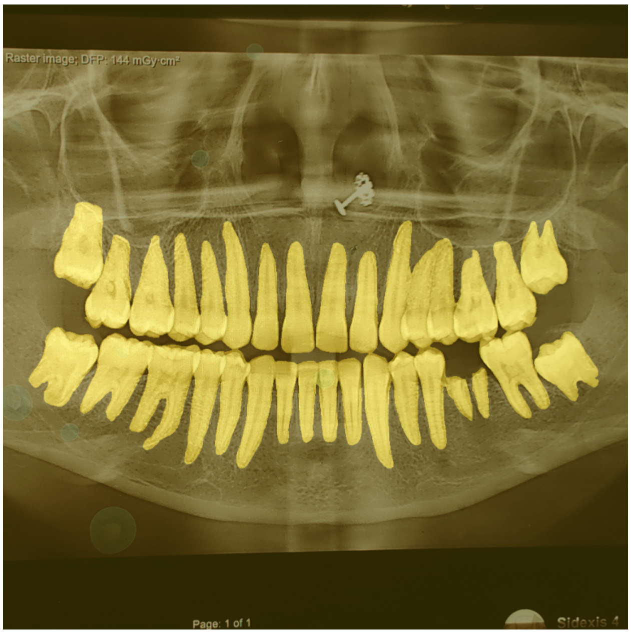


Figure 5.11: Visual Inspection for Residual U-net (three-layer variant)

SE U-net Looking at figure 5.12, we can see that the teeth are clearly defined, and the boundaries between the teeth and gums are accurately captured for most of the parts by SE U-net (two-layer variant). The overall segmentation quality was high for this model with two convolutional layers per block.

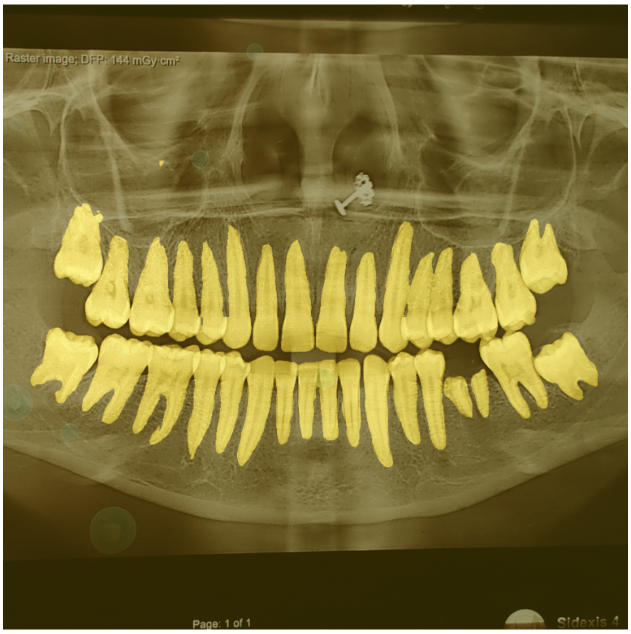


Figure 5.12: Visual Inspection for SE U-net (two-layer variant)

As seen from figure 5.13, the majority of the teeth in the images, including the incisors, canines, and premolars, have been correctly classified and separated by this model (three-layer variant), just like Residual U-net. However, the model appears to have misclassified some of the surrounding tissue as being part of the upper-left wisdom teeth.

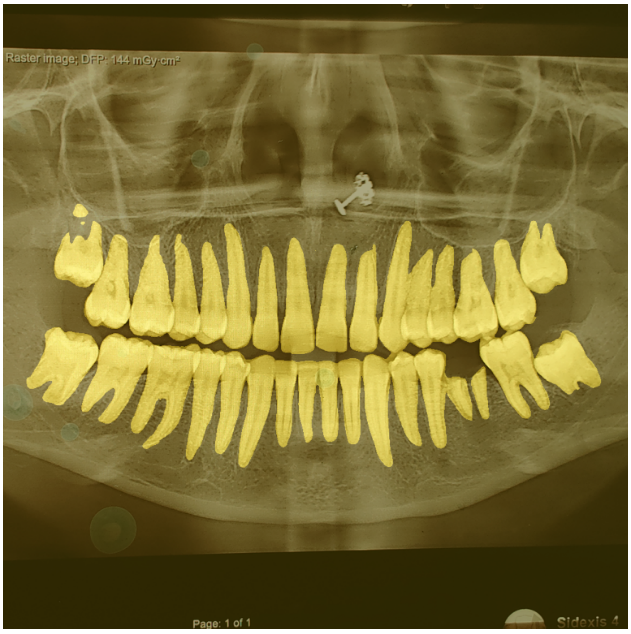


Figure 5.13: Visual Inspection for SE U-net (three-layer variant)

Upon visual inspection of the segmentation results, all the models showed decent performance in terms of accurately identifying the teeth in the test image. Upon closer examination of the results, some issues appeared, such as small regions of the gums being misclassified as teeth. But these cases were small in numbers and overall, the segmentation models performed well, specifically in terms of identifying the main boundaries of the teeth.

5.2 Graphical Analysis

Figure 5.14 provides a visual segmentation accuracy vs. mean epoch train time vs. architecture parameter with a subplot of the four matrices (Accuracy, F1 score, Dice coefficient, and IoU) for our dataset. Here, the circle shapes represent the two-layer variants, and the square shapes represent the three-layer variants. Also, the sizes of the two shapes vary based on their network complexities. As we've seen in the visual inspection analysis, their performance is hard to distinguish. Moreover, it is known that accuracies are crucial factors to take into account when comparing models, but they are not the only ones. Other factors, such as network complexity and train time, should also be taken under consideration, especially if most of the models are producing similar results. So, the idea behind creating this graph is to get a generalized idea about the performance as well as evaluation time and complexity of all the models at a glance.

For all four matrices, we can see that most of the two-layer variants (circles) are clustered together or very close to one another. A similar occurrence can be found among the three-layer variants (squares). This portrays that most of the models with the same variant are conveying comparable performances. On the other, with an increase of convolutional layer per block, all six models seem to improve slightly. But it is the cost of increasing the model complexity to almost 1.5 times for all the architectures. Not only that, as we can see that all of the models' training or evaluation drastically increases when an additional convolutional layer is added per block.

While Dense U-net (three-layer variant) performs best across all matrices, it is the second slowest training U-net variant among all architectures. Moreover, this variant of Dense U-Net uses the most amount of parameters among all the six U-net models. Therefore, one may need to consider this architecture before implementing it due to its model complexity. Similarly, R2 U-net (three-layer variant) is among the best-performing models while being the slowest to evaluate and train. The most impressive among the three-layer variants has to be SE U-net. Not only does it appear to have training time that is relatively comparable to those of the majority of two-layer variations, but it also has one of the best segmentation accuracies.

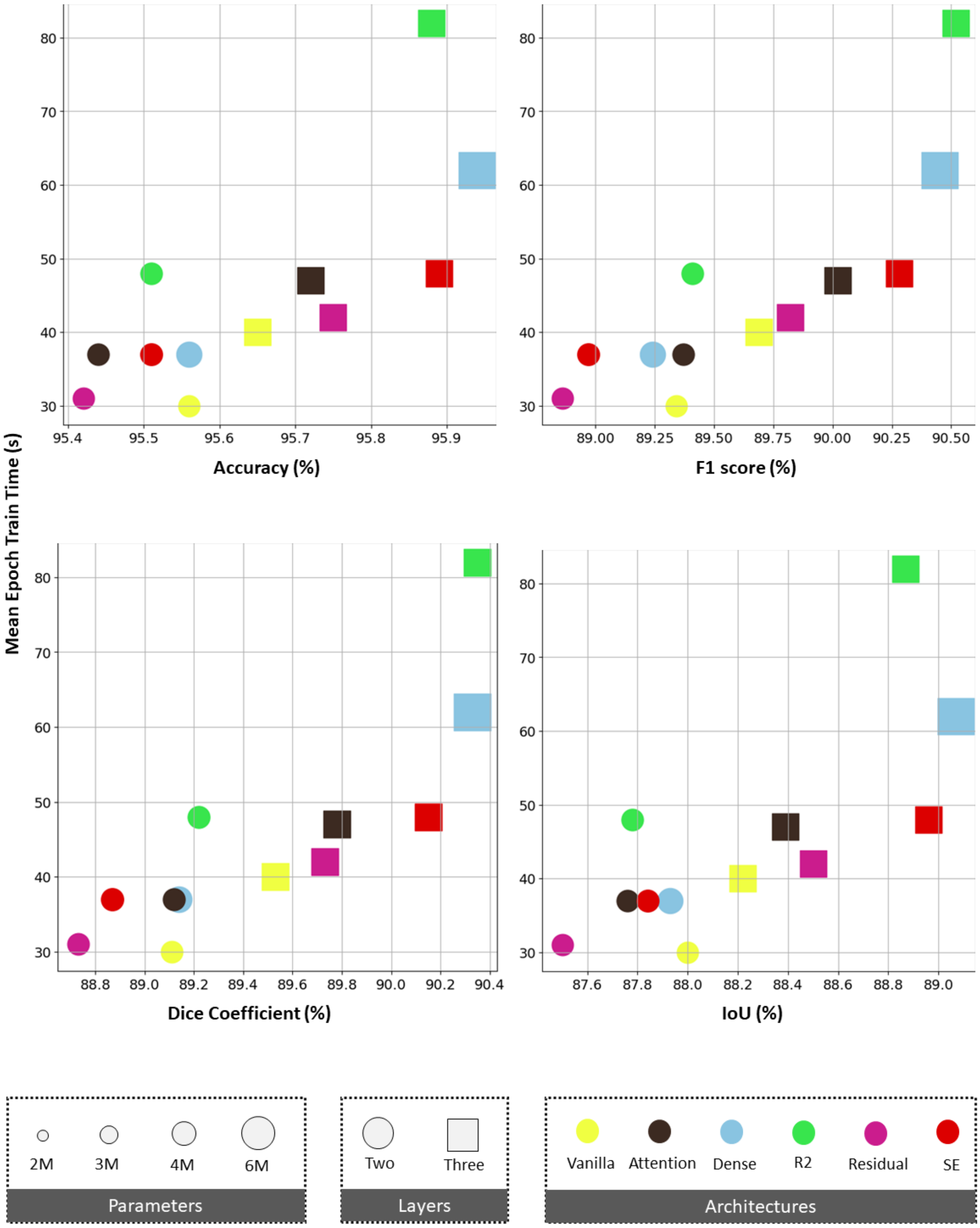


Figure 5.14: Segmentation accuracy (horizontal axis) vs. mean epoch train time (vertical axis) vs. architecture parameters

Considering a faster training model with fewer parameters while excluding the thought of segmentation accuracy, the two-layer variant Vanilla U-net stands out the most. Because not only it possesses the fewest network complexity among the architectures, but also it has the lowest training and evaluation time. Even more impressive is that its performance is very much comparable to all the architectures. While Attention U-net and Residual U-net might not be among the best in segmentation accuracy, their two-layer variants are one of the fastest to train.

These results demonstrate that one or more factors, such as training speed, model complexity, or segmentation accuracy, must be compromised. However, many may come to an agreement that the benefits in performance are negligible across all architectures. Therefore, suggesting faster, simplified architectures seem to be an optimal solution for OPG teeth segmentation.

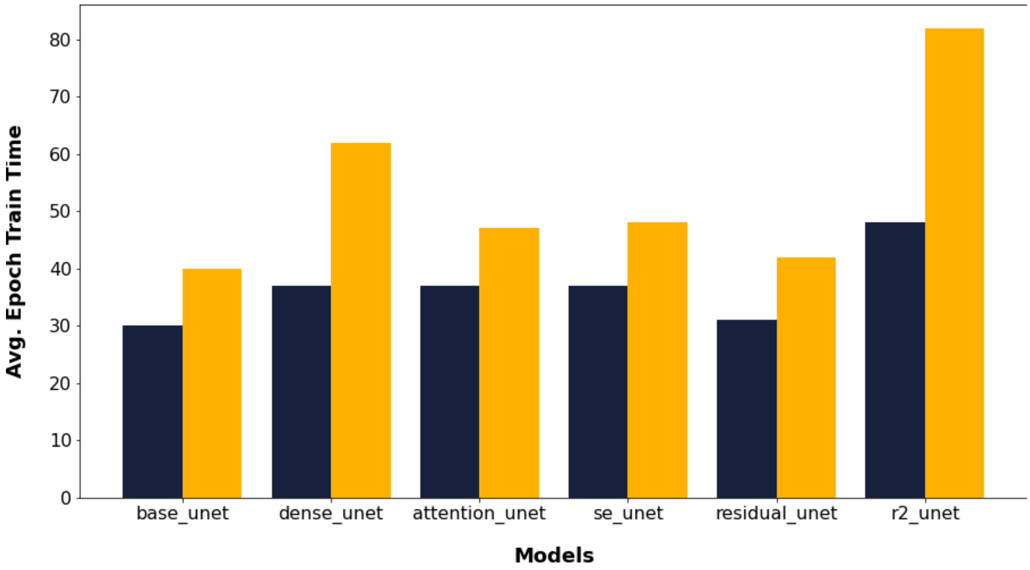


Figure 5.15: Mean epoch training time for each model

Figure 5.15 shows us the differences in the time taken to train the models. As mentioned earlier, Vanilla U-net appears to be the fastest training U-net model, while R2-Unet is the slowest by a significant margin in comparison to all other architectures.

Table 5.1 provides the summary of the results obtained for our dataset. The findings imply that all of the architectures appear to operate in a relatively comparable manner. We can observe that the accuracy (measured using four matrices; Accuracy, F1 score, Dice coefficient, and IoU) of the six U-Net architectures was similar, while each architecture's performance was marginally increased by adding an extra convolutional layer per block (making it three instead of two). The addition of a convolutional layer per block, unfortunately, came at the expense of larger and more complex and training time.

Although each architecture's performance has improved as a consequence, our conclusion on the comparison between them has essentially stayed unchanged. In fact, despite some of the U-Net variations having reduced training speed and higher complexity, their segmentation accuracy was not noticeably better than the standard

Architecture	Conv. Layer	Accuracy (SD) [%]	F1 Score (SD) [%]	Dice Coef. (SD) [%]	IoU (SD) [%]	Mean Epoch Train Time (SD) [s]	Params [x 10^6]
Vanilla U-net	2	95.56(0.01)	89.34(0.08)	89.11(0.08)	88.00(0.05)	30(0.5)	1.9
	3	95.65(0.01)	89.69(0.07)	89.53(0.06)	88.22(0.04)	40(0.53)	2.9
Attention U-net	2	95.44(0.01)	89.37(0.09)	89.12(0.08)	87.76(0.05)	37(0.74)	2
	3	95.72(0.02)	90.02(0.09)	89.78(0.08)	88.39(0.05)	47(0.61)	3
Dense U-net	2	95.56(0.01)	89.24(0.07)	89.14(0.06)	87.93(0.04)	37(0.53)	2.7
	3	95.94(0.01)	90.45(0.06)	90.33(0.06)	89.07(0.04)	62(0.62)	5.4
R2 U-net	2	95.51(0.02)	89.41(0.09)	89.22(0.09)	87.78(0.05)	48(0.81)	2
	3	95.88(0.02)	90.52(0.09)	90.35(0.09)	88.87(0.05)	82(0.92)	3
Residual U-net	2	95.42(0.01)	88.86(0.06)	88.73(0.06)	87.50(0.04)	31(0.52)	1.9
	3	95.75(0.01)	89.82(0.07)	89.73(0.06)	88.50(0.04)	42(0.53)	2.9
SE U-net	2	95.51(0.02)	88.97(0.09)	88.87(0.08)	87.84(0.05)	37(0.61)	2.1
	3	95.89(0.02)	90.28(0.09)	90.15(0.07)	88.96(0.05)	48(1.13)	3

Table 5.1: Comparison of all architectures

Vanilla U-Net, which already performs quite well for OPG teeth segmentation.

In addition to segmentation accuracy, other significant factors, including assessment speed and network complexity, should be taken into account from the standpoint of practical implementations in both clinical settings and research. In some applications, modest performance gains may not be desirable if they come at the expense of much more complexity and slower speed.

Chapter 6

Future Work and Conclusion

6.1 Conclusion

Panoramic x-ray images help to detect diseases that are hardly visible to dentists; manual identification relies entirely on the dentist's expertise. So, better diagnosis models are required to have a better inspection that demands better teeth segmentation from the input panoramic x-ray images, which is highly emphasized in our paper. In this study, we provided an extensive and impartial analysis and comparison of six U-Net architectures of both two and three-layer variants for teeth segmentation of panoramic x-ray radiographs on our dataset, which can aid in creating a successful segmentation model. All the U-Net models used in our study perform significantly well in teeth segmentation from dental panoramic x-rays from our dataset. However, after analyzing the results using the dice coefficient and IoU score as the main accuracy matrices, their performance has minimal differences. Rather depending on the condition of clinical applications and the limitation of time and complexity, a few adjustments are highlighted in this study. Our paper mainly focuses on the impartial and deep analysis of the U-net models that can be very useful for segmentation approaches and significantly impact the ever-changing U-Net models, which can help build disease diagnosis models using the best segmentation method. In this study, we have found that the 3-layer variants of R2 U-net (Recurrent Residual U-net) and the Dense U-net is significantly better in performance with dice coefficient percentage of 90.35 and 90.33, respectively, and with the IoU score of 88.87 and 89.07, respectively. According to our study, focusing only on performance and segmentation accuracy, the R2 U-net and Dense U-net can be the optimum for teeth segmentation on panoramic x-ray images. But considering the complexity of more layers and time, these two variants can be unsuccessful. On the other hand, the 2-layer variant of the Vanilla U-net model with a dice coefficient percentage of 89.11 with less time and layers can be a very optimal model in terms of clinical applications considering limitations and less hardware availability. Our study, which includes impartial analysis and optimal model selection for teeth segmentation, can be significantly helpful in the future findings of the optimal model for segmentation without wasting valuable time in this field. To conclude, our study can determine the comparisons finding the optimum U-net model for teeth segmentation on our dataset made from scratch, which can be significantly valuable in future studies reducing human effort and time in this field of research.

6.2 Future Work

In our work, we have implemented U-Net as our base segmentation model. Additionally, we used several variants of U-Net and achieved satisfactory accuracy. In our future work, we will differentiate between healthy and unhealthy teeth. Then, we will train our model in such a way that our model can detect dental diseases. Last but not least, we will try different deep learning learning algorithms to detect diseases and try to acquire the best possible result.

Bibliography

[1] C. W. Douglass, R. W. Valachovic, A. Wijesinha, H. H. Chauncey, K. K. Kapur, and B. J. McNeil, “Clinical efficacy of dental radiography in the detection of dental caries and periodontal diseases,” en, *Oral Surg. Oral Med. Oral Pathol.*, vol. 62, no. 3, pp. 330–339, Sep. 1986.

[2] E. D. Beltrán-Aguilar, L. K. Barker, M. T. Canto, *et al.*, “Surveillance for dental caries, dental sealants, tooth retention, edentulism, and enamel fluorosis–united states, 1988-1994 and 1999-2002,” en, *MMWR Surveill. Summ.*, vol. 54, no. 3, pp. 1–43, Aug. 2005.

[3] R. J. Manski, J. F. Moeller, H. Chen, J. Schimmel, P. A. St Clair, and J. V. Pepper, “Dental usage under changing economic conditions,” en, *J. Public Health Dent.*, vol. 74, no. 1, pp. 1–12, 2014.

[4] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” *CoRR*, vol. abs/1505.04597, 2015. arXiv: 1505.04597. [Online]. Available: http://arxiv.org/abs/1505.04597.

[5] P. Amrollahi, B. Shah, A. Seifi, and L. Tayebi, “Recent advancements in regenerative dentistry: A review,” en, *Mater. Sci. Eng. C Mater. Biol. Appl.*, vol. 69, pp. 1383–1390, Dec. 2016.

[6] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

[7] M. Z. Alom, M. Hasan, C. Yakopcic, T. M. Taha, and V. K. Asari, “Recurrent residual convolutional neural network based on u-net (r2u-net) for medical image segmentation,” *CoRR*, vol. abs/1802.06955, 2018. arXiv: 1802.06955. [Online]. Available: http://arxiv.org/abs/1802.06955.

[8] A. Ehsani Rad, M. Rahim, H. Kolivand, and A. Norouzi, “Automatic computer-aided caries detection from dental x-ray images using intelligent level set,” *Multimedia Tools and Applications*, vol. 77, pp. 1–20, Nov. 2018. DOI: 10.1007/s11042-018-6035-0.

[9] G. Jader, J. Fontineli, M. Ruiz, K. Lima, M. Pithon, and L. Oliveira, “Deep instance segmentation of teeth in panoramic x-ray images,” Oct. 2018, pp. 400–407. DOI: 10.1109/SIBGRAPI.2018.00058.

[10] O. Oktay, J. Schlemper, L. L. Folgoc, *et al.*, “Attention u-net: Learning where to look for the pancreas,” *CoRR*, vol. abs/1804.03999, 2018. arXiv: 1804.03999. [Online]. Available: http://arxiv.org/abs/1804.03999.

[11] A. G. Roy, N. Navab, and C. Wachinger, "Concurrent spatial and channel squeeze & excitation in fully convolutional networks," *CoRR*, vol. abs/1803.02579, 2018. arXiv: 1803.02579. [Online]. Available: http://arxiv.org/abs/1803.02579.

[12] N. Vila Blanco, I. Tomás Carmona, and M. Carreira, "Fully automatic teeth segmentation in adult opg images," *Proceedings*, vol. 2, p. 1199, Sep. 2018. DOI: 10.3390/proceedings2181199.

[13] L. Fiorillo, "Oral health: The first step to well-being," en, *Medicina (Kaunas)*, vol. 55, no. 10, p. 676, Oct. 2019.

[14] H. Hu, Y. Zheng, Q. Zhou, J. Xiao, S. Chen, and Q. Guan, "Mc-unet: Multi-scale convolution unet for bladder cancer cell segmentation in phase-contrast microscopy images," in *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 2019, pp. 1197–1199. DOI: 10.1109/BIBM47256.2019.8983121.

[15] T. L. Koch, M. Perslev, C. Igel, and S. S. Brandt, "Accurate segmentation of dental panoramic radiographs with u-nets," in *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)*, 2019, pp. 15–19. DOI: 10.1109/ISBI.2019.8759563.

[16] P. Samudre, P. Shende, and V. Jaiswal, "Optimizing performance of convolutional neural network using computing technique," in *2019 IEEE 5th International Conference for Convergence in Technology (I2CT)*, 2019, pp. 1–4. DOI: 10.1109/I2CT45611.2019.9033876.

[17] T. Ahmed, P. Das, M. F. Ali, and M. F. Mahmud, "A comparative study on convolutional neural network based face recognition," in *2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 2020, pp. 1–5. DOI: 10.1109/ICCCNT49239.2020.9225688.

[18] S. Cai, Y. Tian, H. Lui, H. Zeng, Y. Wu, and G. Chen, "Dense-UNet: A novel multiphoton in vivo cellular image segmentation model based on a convolutional neural network," en, *Quant. Imaging Med. Surg.*, vol. 10, no. 6, pp. 1275–1285, Jun. 2020.

[19] J. Charvát, A. Procházka, M. Fričl, O. Vyšata, and L. Himmlová, "Diffuse reflectance spectroscopy in dental caries detection and classification," en, *Signal Image Video Process.*, vol. 14, no. 5, pp. 1063–1070, Jul. 2020.

[20] V. Geetha, K. S. Aprameya, and D. M. Hinduja, "Dental caries diagnosis in digital radiographs using back-propagation neural network," *Health Information Science and Systems*, vol. 8, pp. 1–14, 2020.

[21] A. Haghanifar, M. M. Majdabadi, and S.-B. Ko, "Automated teeth extraction from dental panoramic x-ray images using genetic algorithm," in *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2020, pp. 1–5. DOI: 10.1109/ISCAS45731.2020.9180937.

[22] A. Haghanifar, M. M. Majdabadi, and S.-B. Ko, "Paxnet: Dental caries detection in panoramic x-ray using ensemble transfer learning and capsule classifier," *ArXiv*, vol. abs/2012.13666, 2020.

[23] M. Jafari, D. Auer, S. Francis, J. Garibaldi, and X. Chen, "Dru-net: An efficient deep convolutional neural network for medical image segmentation," in *2020 IEEE 17th International Symposium on Biomedical Imaging (ISBI)*, 2020, pp. 1144–1148. DOI: 10.1109/ISBI45749.2020.9098391.

[24] L. Liu, J. Xu, Y. Huan, Z. Zou, S.-C. Yeh, and L.-R. Zheng, "A smart dental health-IoT platform based on intelligent hardware, deep learning, and mobile terminal," en, *IEEE J. Biomed. Health Inform.*, vol. 24, no. 3, pp. 898–906, Mar. 2020.

[25] F. A. Uçkun, H. Özer, E. Nurbaş, and E. Onat, "Direction finding using convolutional neural networks and convolutional recurrent neural networks," in *2020 28th Signal Processing and Communications Applications Conference (SIU)*, 2020, pp. 1–4. DOI: 10.1109/SIU49456.2020.9302448.

[26] T. M. Wani, T. S. Gunawan, S. A. A. Qadri, H. Mansor, M. Kartiwi, and N. Ismail, "Speech emotion recognition using convolution neural networks and deep stride convolutional neural networks," in *2020 6th International Conference on Wireless and Telematics (ICWT)*, 2020, pp. 1–6. DOI: 10.1109/ICWT50448.2020.9243622.

[27] H. Yu, Z. Lin, Y. Liu, J. Su, B. Chen, and G. Lu, "A new technique for diagnosis of dental caries on the children's first permanent molar," *IEEE Access*, vol. 8, pp. 185 776–185 785, Jan. 2020. DOI: 10.1109/ACCESS.2020.3029454.

[28] X. Qin, M. Xu, C. Zheng, C. He, and X. Zhang, "Multi-scale feedback feature refinement u-net for medical image segmentation," in *2021 IEEE International Conference on Multimedia and Expo (ICME)*, 2021, pp. 1–6. DOI: 10.1109/ICME51207.2021.9428150.

[29] N. Siddique, S. Paheding, C. P. Elkin, and V. Devabhaktuni, "U-net and its variants for medical image segmentation: A review of theory and applications," *IEEE Access*, vol. 9, pp. 82 031–82 057, 2021. DOI: 10.1109/ACCESS.2021.3086020.

[30] F. G. Zanjani, A. Pourtaherian, S. Zinger, *et al.*, "Mask-mcnet: Tooth instance segmentation in 3d point clouds of intra-oral scans," *Neurocomputing*, vol. 453, pp. 286–298, 2021, ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2020.06.145. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231221001041.

[31] J. Im, J.-Y. Kim, H.-S. Yu, *et al.*, "Accuracy and efficiency of automatic tooth segmentation in digital dental models using deep learning," *Scientific Reports*, vol. 12, no. 1, p. 9429, Jun. 2022, ISSN: 2045-2322. DOI: 10.1038/s41598-022-13595-2. [Online]. Available: https://doi.org/10.1038/s41598-022-13595-2.

[32] T. J. Jang, K. C. Kim, H. C. Cho, and J. K. Seo, "A fully automated method for 3d individual tooth identification and segmentation in dental cbct," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 10, pp. 6562–6568, 2022. DOI: 10.1109/TPAMI.2021.3086072.

[33] S. Lin, X. Hao, Y. Liu, D. Yan, J. Liu, and M. Zhong, "Lightweight deep learning methods for panoramic dental x-ray image segmentation," *Neural Computing and Applications*, Dec. 2022, ISSN: 1433-3058. DOI: 10.1007/s00521-022-08102-7. [Online]. Available: https://doi.org/10.1007/s00521-022-08102-7.

[34] C. Sheng, L. Wang, Z. Huang, *et al.*, "Transformer-based deep learning network for tooth segmentation on panoramic radiographs," *Journal of Systems Science and Complexity*, Oct. 2022, ISSN: 1559-7067. DOI: 10.1007/s11424-022-2057-9. [Online]. Available: https://doi.org/10.1007/s11424-022-2057-9.

[35] H. Zhu, Z. Cao, L. Lian, G. Ye, H. Gao, and J. Wu, "Cariesnet: A deep learning approach for segmentation of multi-stage caries lesion from oral panoramic x-ray image," *Neural Computing and Applications*, pp. 1–9, Jan. 2022. DOI: 10.1007/s00521-021-06684-2.