NORTH SOUTH UNIVERSITY

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

CSE 498R Report

Tooth Decay Detection Using Deep Learning Techniques

Nazmul Hasan – 1911742042

Samya Sunibir Das – 1911563042

February, 2023

Declaration

We certify that the work presented in this report, entitled "Tooth Decay Detection Using Deep Learning Techniques," is a result of our independent research and investigation conducted under the guidance of Mr. Rifat Ahmed Hassan, Lecturer in the Department of ECE at NSU. We affirm that this thesis and any part thereof has not been previously submitted for any academic award, diploma, or other qualifications. Any materials utilized in this project have been properly cited.

Student's Names and Signatures:

1. Samya Sunibir Das

2. Nazmul Hasan

Dhaka, Bangladesh

February, 2023

Approval

The report titled "**Tooth Decay Detection Using Deep Learning Techniques**", submitted by Samya Sunibir Das and Nazmul Hasan, in the Fall 2022 session, to the Department of Electrical and Computer Engineering, North South University, has been accepted as satisfactory in partial fulfillment of the requirements for the degree of Bachelor of Science in Electrical and Computer Engineering and approved as to its style and contents on Sunday 12th February, 2023.

Supervisor's Signature:

____

Rifat Ahmed Hassan

Lecturer

Department of Electrical and Computer Engineering

North South University

Dhaka, Bangladesh.

Department Chair's Signature:

____

Dr. Rajesh Palit

Professor and Chair

Department of Electrical and Computer Engineering

North South University Dhaka, Bangladesh.

Dedication

We dedicate this work to the teachers, researchers, and our loved ones who have inspired and supported us throughout our academic journey. Their dedication, guidance, and encouragement have been the foundation of our success. This work is a testament to their influence and is a humble tribute to their unwavering support.

TABLE OF CONTENTS

TABLE OF FIGURES

ABSTRACT

Cavities, commonly known as tooth decay, are portions of the hard surface of teeth that have been irreversibly damaged. It eventually turns into tiny gaps or holes. It can cause infection, pain, and even tooth loss if not treated appropriately. For developing and underdeveloped countries, regular dental appointments can be costly. The purpose of this study was to develop and evaluate the performance of a tooth decay detection system that used a deep learning technique based on a convolutional neural network (CNN) to detect tooth decay from oral photographs. We used a collection of 233 pictures of teeth with cavitation. The input was clear photos of affected teeth on a white background. The dataset was trained on three separate object detection models – YOLOv4, YOLOv5m, and YOLOv5s – after some pre-processing. On a pre-trained YOLOv4 model (trained in MS COCO dataset), we used the transfer learning technique to detect dental decay and attained an accuracy of 94.17 percent. YOLOv5s and YOLOv5m, on the other hand, were trained from scratch and had a 97.8% and 96.9% accuracy rate.

Index Terms—Dataset, tooth decay detection, deep learning, transfer learning.

CHAPTER 1: INTRODUCTION

Tooth decay is one of the most widespread health problems in the world. It occurs when a tooth's enamel is damaged. Cavities in the teeth caused by tooth decay can lead to tooth loss. According to a recent study, tooth decay has surpassed heart disease as the most frequent health problem worldwide, affecting almost 34.1% of the population [2]. People in many parts of the world have limited access to dental specialists. Since caries is not life threatening, many patients with untreated caries wait until it is too late, when major complications have already occurred and treatment is too expensive. Dental cavities that go untreated can lead to pulpitis and periapical disorders [3]. However, tooth decay can be halted and the decay process reversed if diagnosed early enough. The enamel has a self-healing property. As a result, early detection of tooth decay is an important factor of treatments to prevent dental caries. It may make dental treatment more affordable for persons with lower and medium incomes.

Artificial Intelligence (AI) is becoming increasingly popular and widely used in medicine to diagnose and treat patients more rapidly and accurately. Deep learning (DL), an artificial intelligence (AI) method, has been used to automate decision-making processes in numerous clinical dental situations in recent years [4]. By autonomously learning from datasets containing human annotations from dental specialists, the approach, which consists of multilayer ConvNets, has already showed promising accuracy on unforeseen data.

The focus of this study was on identifying tooth decay in three phases. The three phases are visible change without cavitation, visible change with micro-cavitation, and visible change with cavitation. Each phase is distinct in terms of personality, patterns, and shapes. The characteristics of the stages are described as follows and illustrated in Figure 1:

- ***Visible change without cavitation:*** This is the earliest stage of tooth decay, when a lesion forms on the tooth. It causes a slight darkening of the tooth's surface, which is usually white or brown [1].
- ***Visible change with micro-cavitation:*** In this stage, demineralization continues and the tooth enamel (the uppermost layer of the tooth's structure) begins to break down.

- ***Visible change with cavitation:*** In this stage, the dentin layer of the tooth is impacted as the tooth decay proceeds. Bacteria get inside the decaying pulp and causes infection.

Figure 1: (a) Visible change without cavitation, (b) Visible change with micro-cavitation, (c) Visible change with cavitation

This paper offers a method for accurately predicting and classifying the three phases of tooth decay using deep learning techniques. For the automatic detection of dental decay from oral photos, we constructed a deep Convolutional Neural Network. The model classifies the presence of dental decay in a given image and uses bounding boxes to locate the findings.

We used three different YOLO object detection models to train the dataset. In section V, the comparison study of the three has been analyzed for a better representation and understanding of the trained model's efficiency and accuracy. The rest of the paper is set out as follows: The second section covers relevant works. Section III dives into the model's training phases. Experimental setup is covered in Section IV. Section VI concludes with some ideas for the future.

CHAPTER 2: RELATED WORK

Agnes Holtkamp and colleagues [5] aimed to train a deep convolutional neural network (CNN) to detect caries lesions using Near-Infrared Light Trans illumination (NILT) imaging obtained in vitro or in vivo, as well as to assess the models' generalizability. 226 extracted permanent posterior human teeth were fitted in a diagnostic model in a dummy head in vitro. The pictures were then segmented tooth-wise using NILT. 1319 teeth from 56 patients were taken in vivo and segmented in the same way. They trained both the vivo and vitro dataset on ResNet [7] classification model. The authors employed a 10-fold cross validation technique on their dataset to improve the accuracy of evaluating a predictive model. To improve explainability, GrandCAM [8] was used. In vivo data models performed much better (mean ± SD accuracy: 0.78 ± 0.04) than in vitro data models (accuracy: 0.64 ± 0.15; p 0.05).

A higher accuracy was found in paper [6] where the authors aimed to present a computational technique that can automatically recognize carious lesions on tooth occlusal surfaces in smartphone photos according to the International Caries Detection and Assessment System (ICDAS). 620 unrestored molars/premolars were photographed in smartphone. The images were evaluated with ICDAS II and classified the images into three classes for caries diagnosis: "No Surface Change," "Visually Non-Cavitated," and "Cavitated" (C). The classifications "C vs (VNC+NSC)" and "VNC versus NSC" were given as two-step detection techniques using Support Vector Machines (SVM). The accuracy, sensitivity, and specificity of "C vs (VNC+NSC)" were 92.37%, 88.1%, and 96.6%, respectively, while "VNC versus NSC" were 83.3%, 82.2%, and 66.7%, respectively.

Deep learning techniques were also used in [9], for screening dental caries from oral photographs. The authors developed and evaluated the performance of a deep learning system based on convolutional neural network (ConvNet) by adopting from Single Shot MultiBox Detector to detect dental caries from images. The classification area under the curve (AUC) of the system was 85.65%. (95% confidence interval: 82.48 % to 88.71%). At a high-sensitivity operating point, the model attained an image-wise sensitivity of 81.90 % and a box-wise sensitivity of 64.60%.

CHAPTER 3: PROPOSED METHODOLOGY

The major objective of this work is to develop tooth decay detection model that uses deep learning techniques to aid tooth decay identification. The approach adopted in this work is outlined in Figure. 2.

A. BLOCK DIAGRAM

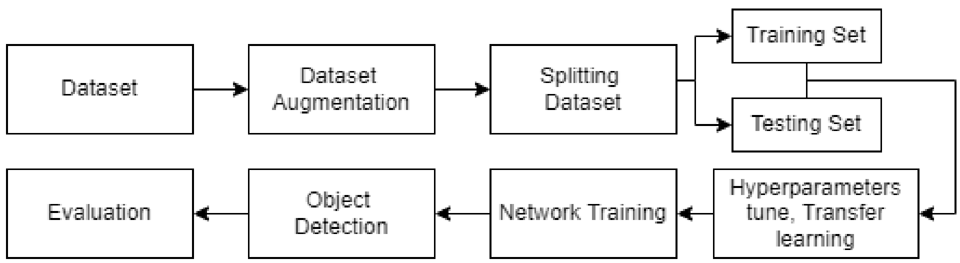

Figure 2: Block Diagram of the entire work

The process of the entire project is depicted in Figure 2. We started by preprocessing the dataset. The dataset was separated into train and test sets after preprocessing. To develop the object detection model, DL techniques were used on the training set. The test set was then used to assess the models' performance.

B. DATASET

1. Dataset Acquisition: Teeth photos with cavitation, micro-cavitation, and no cavitation were collected from several smartphones to form the tooth decay dataset. There are 233 pictures of teeth in varying stages of decay in the dataset. The dataset was manually annotated by human annotators and encoded in MS COCO format [11].

2. Dataset Augmentation: We did data augmentation on the existing dataset to increase training and validation accuracy because the dataset only contains 233 photographs. Six augmentation techniques: vertical flip, horizontal flip, 90° rotation, horizontal-vertical flip, average blurring, and raise hue are utilized to turn the dataset into

1631 photos. YOLOv4 [12] and YOLOv5m are trained using this augmented dataset. The images in the dataset were resized to 416x416 pixels for training in YOLOv5s.

3. Removing Null Values: The dataset was uploaded to Roboflow [10] for image augmentation before training on YOLOv5s. Roboflow found 475 images with Null values after augmentation. Those were taken out of the dataset.

4. Dataset Splitting: For YOLOv4 and YOLOv5m, the dataset was split into two portions for our study: training and validation sets, with the training set containing 1468 instances (90 % of 1631) and the validation set containing 163 instances (10 % of 1631). The dataset for YOLOv5s was divided into three parts: a training set with 86% data, a validation set with 9% data, and a test set with 5% data

C. Model Training

The dataset was trained in three YOLO object detection models. Chosen models are: YOLOv4, YOLOv5s, and YOLOv5m. Model descriptions and how the dataset is trained on the model are described as follows:

YOLOv4: The YOLOv4 algorithm is an enhanced version of the YOLOv3 method. It connects the YOLOv3 head with a CSP darknet-53 [12] classifier and spatial pyramid pooling. It benefits from excellent detection accuracy, precise bounding box positioning, and fast computations. YOLOv4 takes an image as input and compresses features using a backbone of convolutional neural networks. These backbones represent the network's endpoint in picture categorization, and they can be used to make predictions. YOLOv4 is pretrained on ImageNet classification [13]. We used the pretrained YOLOv4 weights and used it to train the model on our custom dataset via transfer learning [14].

Activation Function: Non-monotonic, smooth activation function Mish was chosen due to its minimal cost and qualities such as unbounded above and below, which boost its performance when compared to other often used functions. The Mish function has the following definition:

$$f(x) = x.\tanh\big(s(x)\big)\,;$$

where $s(x) = \ln(1 + e^{x})$, is a softmax activation function.

YOLOv5: YOLOv5 is available in four different models. Except for the model layer architecture and a few parameters, there are no significant differences between the versions [15]. We chose YOLOv5s and YOLOv5m as we had a limited dataset and trained the models from scratch to detect tooth decay. These two models were trained in two different settings to see which one performed better on our custom object detection. Stochastic Gradient Descent (SGD) optimization was used and the learning rate was 0.01.

Activation Function: YOLOv5 uses Leaky ReLU activation function for hidden layers. The use of Leaky ReLU reduces the computation necessary to drive the neural network from growing exponentially. It has the following definition:

f(y)=(αy) if (y<0)

f(y) = y if y>=0 where α is a small constant

Sigmoid activation function that was used in the final detection layer. It transforms the input into a value between 0 and 1. The function is defined as follows:

$$\text{Sigmoid } s(x) = \frac{1}{1+e^{-x}}$$

Loss Function: Binary cross entropy loss function was used. Each of the projected probabilities is compared to the actual class output, which can be either 0 or 1. The score is then calculated, penalizing the probabilities based on their deviation from the predicted value.

$$B.C.E = -\frac{1}{N}\sum_{i=1}^{N}[y_i * \log\big(y_{pred}\big) + (1 - y_i)\& * \log(1 - y_{pred})]$$

Where y_{pred} is the i^{th} scalar value in the model output and y_i is the corresponding target value.

CHAPTER 4: EXPERIMENTAL SETUP

After classifying the dataset into three classes: 0 (visible change without cavitation), 1(visible change with micro-cavitation), and 2(visible change with cavitation), the dataset was fed into the model as shown in Fig. 2. We employed three YOLO object detection models. The experimental setup for YOLOv4, YOLOv5s, and YOLOv5m is described below-

YOLOv4: Images are fed via convolutional down sampling, then supplied through a succession of layers of dense connection blocks that execute various operations and calculations. The outputs of these blocks were then routed via a spatial pyramid pooling layer to widen receptive fields, and then through an object identification layer to identify the various classes in a picture.

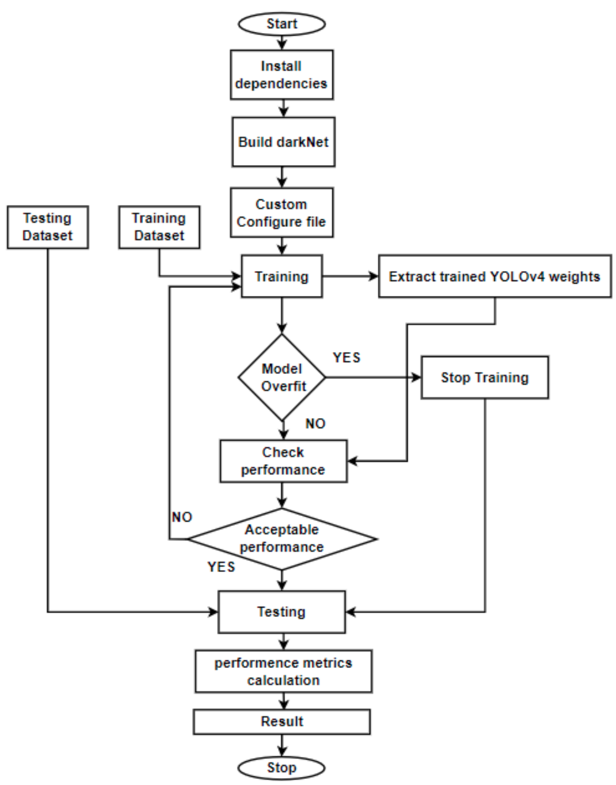

Figure 3: Flowchart of implementation

Figure. 3. Shows the detailed flowchart of design and implementation from splitting dataset to evaluating result. All the necessary dependencies such as YOLOv4, CUDA, NumPy, and Python were installed from respective repositories. DarkNet [12] is a CUDA-based open-source neural network framework designed to support graphics processing units (GPU). A custom configuration file was created from the cloned YOLOv4 repository to build a custom object detector for tooth decay detection. All hyper parameters designed in the development of the custom object detector were detailed in the custom configuration file. The training module was then incorporated with a specific configuration file. Then, the model became ready to be trained with a custom dataset.

Design Constraints and Parameters-Image Dataset

- Number of Classes: 3
- Class name: Visible change without cavitation, Visible change with microcavitation, Visible change with cavitation.
- Filter Size: 416*416
- Batch Size: 64
- Subdivision: 32
- Number of filters: (3+5) *3 = 24

Hyper parameters design

- Image Size: 416*416
- Image channels: 3
- Kernel Size: 3*3
- Activation Function: Mish
- Batch Size: 64
- Max batches: 3*2000 = 6000
- Learning rate: 0.001

The trainer's supervision was necessary for each epoch's parameter values, such as mAP, Accuracy, and Precision. To avoid model overfitting, training should be stopped after the given parameter values become constant or have very few changes. After the model was trained, best YOLOv4 weights are extracted which acts as a reference while testing the model on custom testing dataset.

YOLOv5: A similar working flow as seen in Fig. 3 was used to train the dataset in YOLOv5s and YOLOv5m. A dataset of 2618 photos was used to train YOLOv5s, with 86% used for training, 9% for validation, and 5% for testing. The YOLOv5m model, on the other hand, was trained using the same dataset as the YOLOv4 model. The dataset contains 1631 photos, with 90% of them being used for training and 10% for validation. Both datasets were uploaded into Roboflow [10].

Both the model development started by installing necessary dependencies such as Python, Pytroch, YOLOv5, CUDA and roboflow from respective repositories. Datasets were uploaded and then exported into the YOLOv5 PyTorch format, which generated api keys for each dataset. It's worth mentioning that the Ultralytics solution requires a YAML file that defines where your training and test data should be stored. The Roboflow export also creates this format for us. 0 indicates visible changes without cavitation, 1 indicates visible changes with micro-cavitation, and 2 indicates visible changes with cavitation in the data.yaml file. Training configuration for YOLOv5s and YOLOv5m:

Table I: YOLOv5 MODEL CONFIGURATIONS

Required Argument	**YOLOv5s**	**YOLOv5m**
Image Size	416x416	640x640
Batch Size	16	88
Epoch	100	300

Training losses and performance data was saved to a log file created before with the —name flag when the model was trained. The weight values were saved in .pt files. On test photos, we applied the best weights and got somewhat improved accuracy in YOLOv5s.

CHAPTER 5: RESULTS AND ANALYSIS

Precision and recall are two metrics used to evaluate classification and retrieval systems' performance. Precision is the percentage of relevant occurrences among all retrieved examples. Recall, also known as sensitivity, is the percentage of recovered instances among all appropriate models. In a perfect classifier, precision and recall are both one. Object detection models like YOLO use the assessment measure mAP (mean Average Precision). To calculate mAP, we needed IOU, Precision, Recall, Precision Recall Curve, and AP [16][17]. The confidence threshold was used to calculate the mAP.

Definition for mAP (mean average precision)

$$recall = \frac{retrieved\ and\ relevant\ objects}{all\ retrieved\ objects} = \frac{Tp}{Tp + Fp}$$

$$precision = \frac{retrieved\ and\ relevant\ objects}{all\ retrieved\ objects} = \frac{Tp}{Tp + Fp}$$

$$average\ precision\ AP = \sum_{k=0}^{k=n-1} [Recall(k) - Recall(k+1) * Precision(k)]$$

$$mAP = \frac{1}{n} \sum_{k=1}^{k=n} AP\ (k)$$

Where AP_K = the AP of class k and n = number of classes

The YOLOv4 model training took more time compared to YOLOv5s and YOLOv5m. The model achieved the overall mAP of 94.17%. Table II shows the overall result on YOLOv4 training.

Table II: CLASSWISE RESULT ON YOLOv4 BEST WEIGHTS

Class	Average Precision	Recall	mAP@0.50
Visible change without cavitation	99.67%	93.2%	94.17%
Visible change with micro-cavitation	99.64%	94.03%	
Visible change with cavitation	83.2%	88%	

For IOU threshold = 50%, used area under curve for each unique Recall, mean average precision was 0.941691 or 94.17%. The precision, recall and F-1 score for conf_thresh = 0.25 was 0.93, 0.95 and 0.94 respectively.

The YOLOv5s model was trained with 100 epoch and outperformed the other two models. The model achieved the precision of 97.8% on overall class. mAP@.5 was 98.9% and mAP@.5:.95 was 76.7%. Table III shows the overall result achieved in YOLOv5s.

Table III: CLASSWISE RESULT ON YOLOv5s

Class	Precision	Recall	mAP@.5	mAP@.5:.95
All	97.8%	98%	98.9%	76.7%
Visible change without cavitation	99.3%	95.8%	98%	72.4%
Visible change with microcavitation	95.3%	98.2%	99.1%	76.1%
Visible change with cavitation	98.9%	100%	99.6%	81.6%

The model YOLOv5m on the other hand, was trained with 300 epoch and performed slightly less compared to YOLOv5s. The model achieved an accuracy of 96.9%. Table IV shows the overall result achieved in YOLOv5m.

Table IV: CLASSWISE RESULT ON YOLOv5m

Class	Precision	Recall	mAP@.5	mAP@.5:.95
Visible change without cavitation	94.7%	93%	93.6%	75.2%
Visible change with micro-cavitation	97.4%	92%	96.5%	74.7%
Visible change with cavitation	98.6%	96.7%	99.1%	94.6%

A graphical comparison of YOLOv5s and YOLOv5m is shown below,

RESULT OF TRAINING GRAPH:

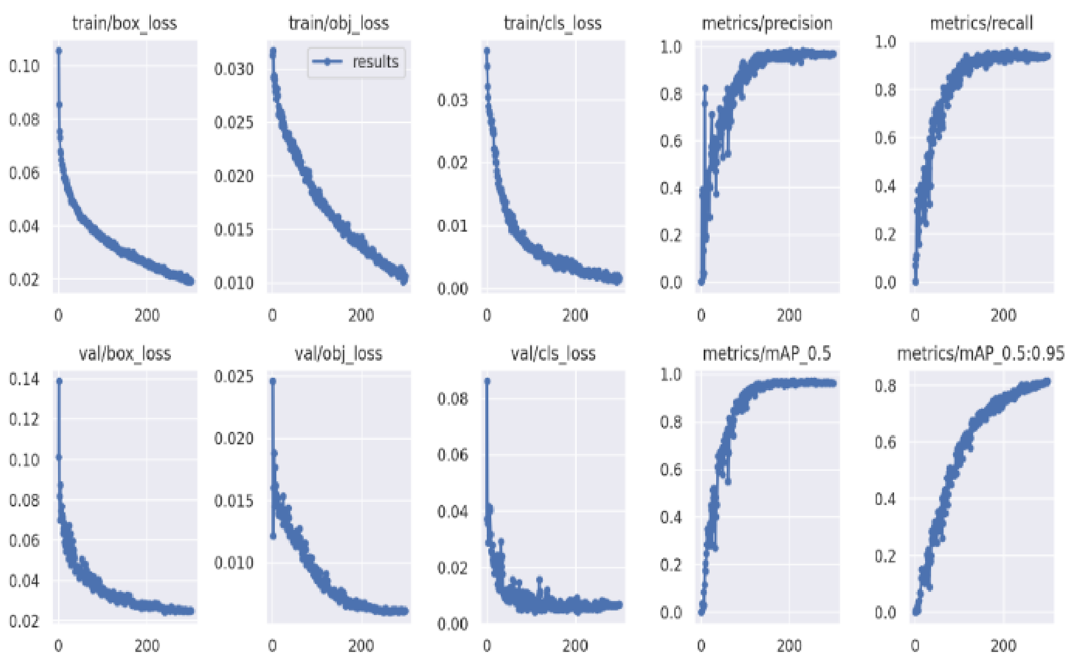

Figure 4: Result of training in YOLOv5m

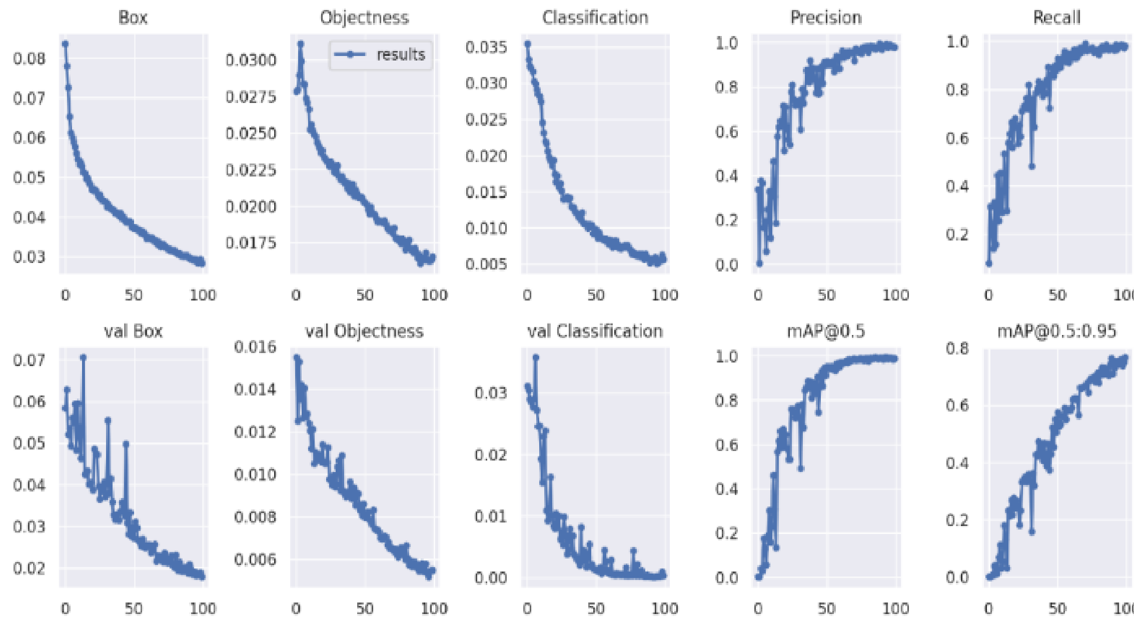

Figure 5: Result of training in YOLOv5s

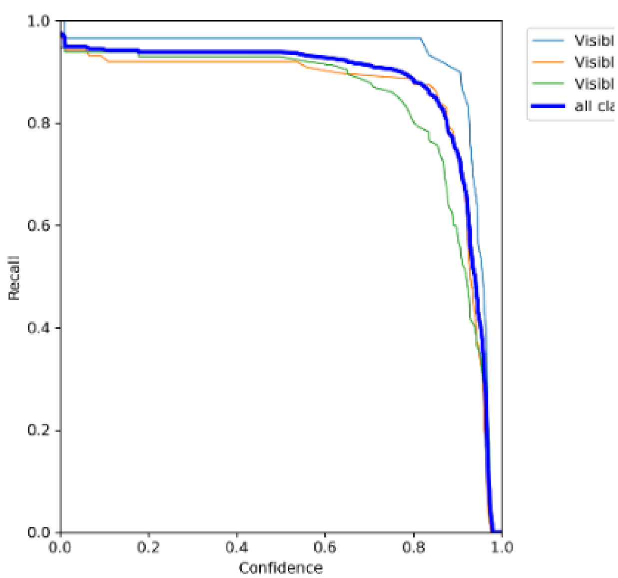

Figure 6: Recall curve for YOLOv5m

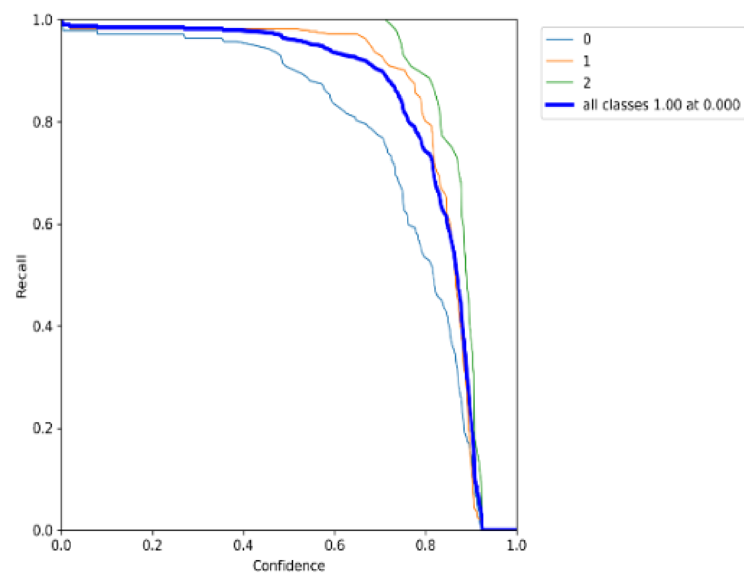

Figure 7: Recall curve for YOLOv5s

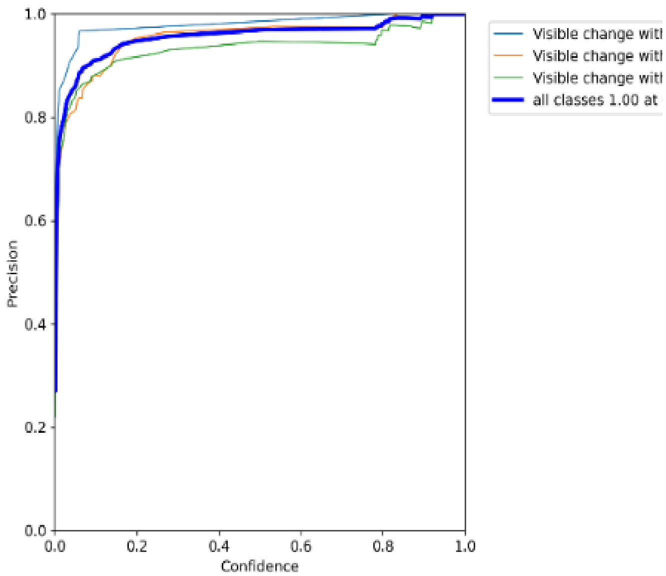

Figure 8: Precision curve for YOLOv5m

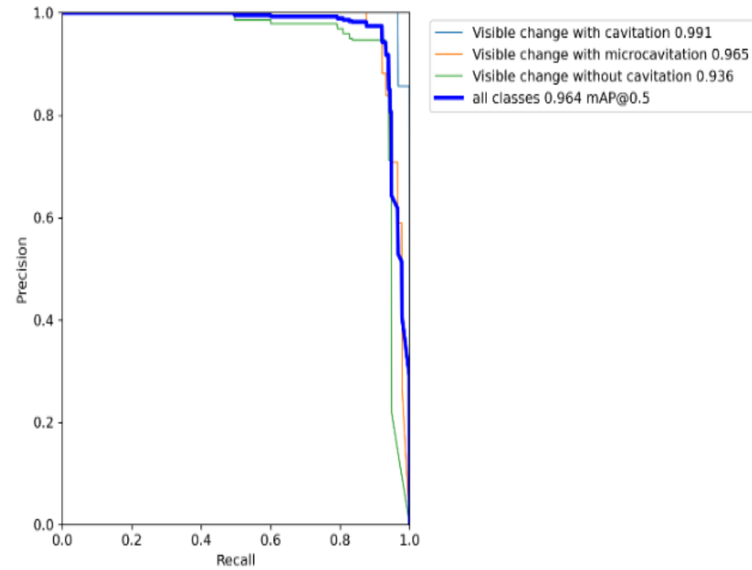

Figure 10: Precision-Recall curve for YOLOv5m

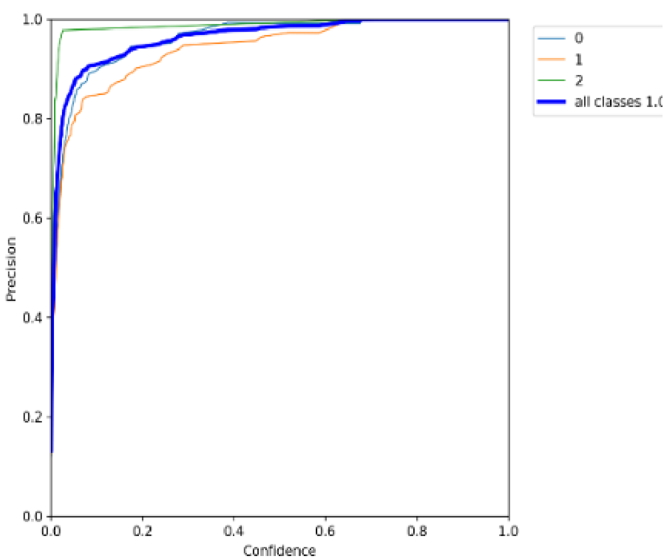

Figure 9: Precision curve for YOLOv5s

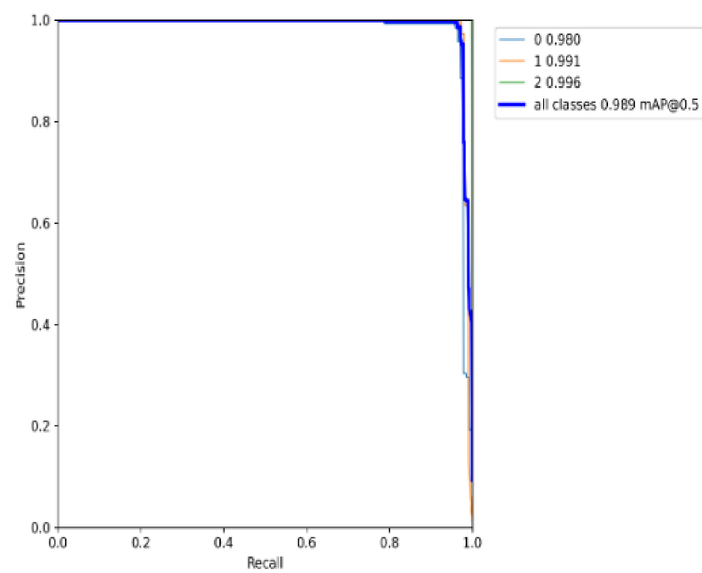

Figure 11: Precision-Recall curve for YOLOv5s

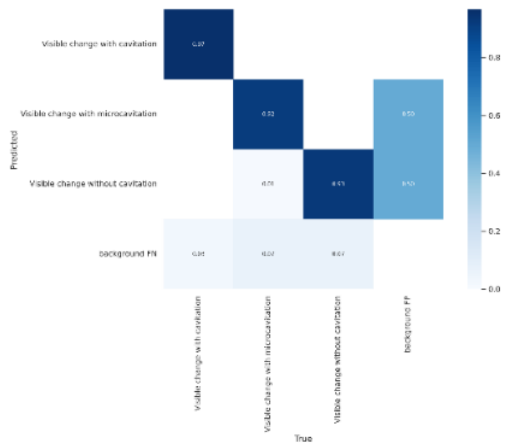

Figure 12: Confusion matrix for YOLOv5m

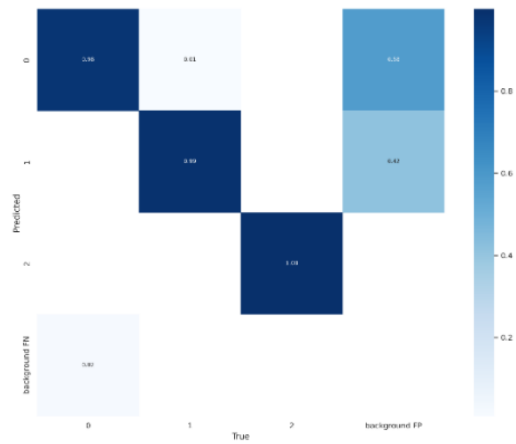

Figure 13: Confusion matrix for YOLOv5s

A sample YOLOv5s model output, in which the model successfully detected all three classes.

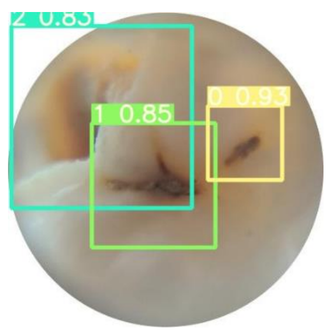

Figure 14: Sample Output of YOLOv5s model

CHAPTER 6: CONCLUSION

Early identification of tooth decay can save costs on dental treatments and even reverse the decay process. A deep learning strategy to detect tooth decay is presented in this paper. The dataset was trained in two different object detection algorithms to see which one performed better. YOLOv5s and YOLOv5m were faster and more accurate to train than YOLOv4. The YOLOv5s model had the highest accuracy of 97.8%, while the YOLOv4 model had the lowest accuracy of 94.17%.

REFERENCES

[1] B. J. N. D.D.S., "The Stages of Tooth Decay: Progression of a Cavity," 16 6 2017. [Online]. Available: https://www.nordhusdentistry.com/blog/2017/06/16/the-stages-of-tooth-decay-183529. [Accessed 12 5 2022].

[2] Peres, M. A., Macpherson, L. M. D., Weyant, R. J., Daly, B., Venturelli, R., Mathur, M. R., Listl, S., Celeste, R. K., Guarnizo-Herreño, C. C., Kearns, C., Benzian, H., Allison, P., & Watt, R. G. (2019). Oral diseases: A global public health challenge. The Lancet, 394, 249–260. https://doi.org/10.1016/S0140-6736(19)31146-8

[3] Ali, A. H., Koller, G., Foschi, F., Andiappan, M., Bruce, K. D., Banerjee, A., & Mannocci, F. (2018). Self-limiting versus conventional caries removal: A randomized clinical trial. Journal of Dental Research, 97(11), 1207–1213. https://doi.org/10.1177/0022034518769255

[4] Park, W., & Park, J. (2018). History and application of artificial neural networks in dentistry. European Journal of Dentistry,12(4),594-601, https://doi.org/10.4103/ejd.ejd_325_18

[5] Agnes Holtkamp, Karim Elhennawy, José E. Cejudo Grano de Oro, Joachim Krois, Sebastian Paris and Falk Schwendicke "Generalizability of Deep Learning Models for Caries Detection," Journal of Clinical Medicine, pp. 6-14, 2021. https://doi.org/10.1177/14604582211007530

[6] Duc Long Duong, Malitha Humayun Kabir, Rong Fu Kuo "Automated caries detection," Journal of Health Informatics Journal, pp. 2-6, November 2021. https://doi.org/10.3390/jcm10050961

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun " Deep Residual Learning for Image Recognition" pp. 3-8. 10 December 2015, https://doi.org/10.48550/arXiv.1512.03385

[8] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, Dhruv Batra " Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization" pp. 4-7, 2016, https://doi.org/10.48550/arXiv.1610.02391

[9] Xuan Zhang, Yuan Liang, Wen Li, Chao Liu, Deao Gu, Weibin Sun, Leiying Miao " Development and evaluation of deep learning for screening dental caries from oral photographs", pp. 2-7, November 2020, DOI: 10.1111/odi.13735

[10] Docs.roboflow.com. 2022. Image Augmentation - Roboflow. [online] Available at: https://docs.roboflow.com/image-transformations/image-augmentation [Accessed 12 May 2022].

[11] Tsung-Yi Lin Michael Maire Serge Belongie Lubomir Bourdev Ross Girshick James Hays Pietro Perona Deva Ramanan C. Lawrence Zitnick Piotr Dollar, " Microsoft COCO: Common Objects in Context", pp. 4-6, 2014, https://doi.org/10.48550/arXiv.1405.0312

[12] Alexey Bochkovskiy, Chien-Yao Wang, Hong-Yuan Mark Liao, " YOLOv4: Optimal Speed and Accuracy of Object Detection", pp. 6-9, 2020, https://doi.org/10.48550/arXiv.2004.10934

[13] "YOLOv4- An explanation of how it works", Roboflow Blog,2022. [Online]. Available: https://blog.roboflow.com/a-thorough-breakdown-of-yolov4/. [Accessed: 14- May- 2022].

[14] "Custom Object Detection with transfer learning with pre-trained YOLO-V4 model", sandipanweb, 2022. [Online].Available:https://sandipanweb.wordpress.com/2022/01/17/custom-object-detection-with-transfer-learning-with-pre-trained-yolo-v4-model/. [Accessed: 14- May- 2022].

[15] "YOLOv5 is Here", Roboflow Blog, 2022. [Online]. Available:https://blog.roboflow.com/yolov5-is-here/. [Accessed: 14- May- 2022].

[16] "IoU a better detection evaluation metric", Medium, 2022.[Online].Available:https://towardsdatascience.com/iou-a-better-detection-evaluation-metric- [Accessed: 14- May- 2022].

[17] "Mean Average Precision (mAP) Explained | Paperspace Blog", Paperspace Blog, 2022. [Online]. Available:https://blog.paperspace.com/mean-average-precision/. [Accessed: 14- May- 2022].