# Dental Caries Detection Using Faster R-CNN and YOLO V3

*Aayush* Juyal[1*], *Himanshu* Tiwari[1], *Ujjwal Kumar* Singh[1], *Nitin* Kumar[1] and *Sandeep* Kumar[1]

[1]Departement of Computer Science and Engineering, Sharda University 203101, Greater Noida, India

**Abstract.** Deep learning techniques are gradually being utilized in many fields. Healthcare is a field in which deep learning can thrive. The study conducted focuses on using deep learning object detection models to detect dental cavities in an individual's mouth. These images taken from a camera will be fed live to the object detection model to discover the precise coordinates of dental caries if it happens to exist. Previous studies depict that X-rays were often used for detecting dental caries. This study wants to put emphasis on avoiding the use of X-rays since they have a chance of harming human tissue, as well as, and they cannot detect hidden caries. Thus, it is necessary to detect dental caries in an accurate manner, with the proper tools. Studies have also conducted dental caries prediction using the frontal view of the images only. Some have made use of different angles for the images in the dataset, however, there still lies the problem of capturing the posterior teeth. Roughly 300 images get used, as the dataset, for the training and testing of the object detection model. 80% is used for training whereas 20% is used for testing. Two deep learning frameworks have been proposed to evaluate dental cavities, the You Only Once (YOLO) V3 object detection model and the Faster Region-Convolutional Neural Network object detection model. Our results show that the YOLO V3 model consists of an accuracy of 75%, while Faster R-CNN had an accuracy of 80%. The sensitivity values of YOLO V3 and Faster R-CNN were 76% and 73% respectively. The model with better performance would be used for future development of the product, along with the hardware components. Our hardware components aim to take images from outside the mouth, for the frontal teeth, and take images from inside the mouth, for the posterior teeth.

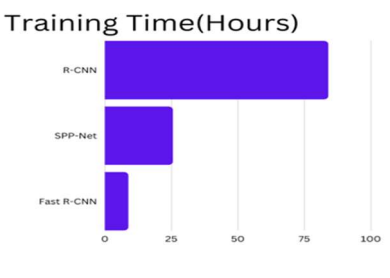
## 1 Introduction

Many people often overlook dental care. Without going to the dentist, detecting any caries or issues related to the teeth is very intricate. Dental caries concludes to tooth pain and tooth

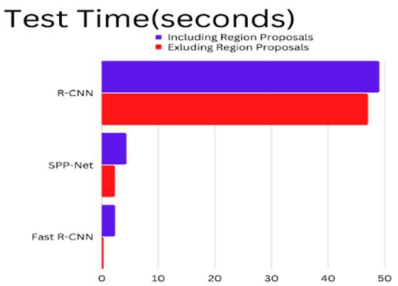
*Corresponding author : aayushjuyal12@gmail.com

loss. It is significant to detect dental caries early on, in order to prevent it from aggravating. According to WHO (World Health Organization), dental diseases are costly to get treated, they take up about 5-10% of the healthcare budget. The normal procedure for the diagnosis of dental caries is through the use of an x-ray, visual inspection, a dental probe, and handheld mirrors [2]. However, an X-ray is found to be harmful and should be avoided [19]. Even though exposure to radiation is minimum, continuous exposure can have a long-term effect by damaging the molecular structure and causing harm. Another issue with the use of X-rays is that they cannot detect 'hidden carries' (invisible at the surface of teeth) in an accurate manner [19]. Finally, much of this procedure is manual, making it laborious for dentists. Deep learning is a vast topic that has taken over almost every field. It is a subset of machine learning, in which the machines have to learn from the given data themselves, with the minimum human intervention [21]. Deep learning is based on the idea of artificial neural networks. A neural network is described as a paradigm that teaches machines to learn and process information in a manner that is inspired by the human brain [7]. This model will serve the purpose of taking as an input, an image of teeth, and then being able to detect where dental caries are located. There are several object detection models today, such as CNN, Region Based CNN(R-CNN), Fast R-CNN, Faster R-CNN, YOLO V3, RetinaNet and etc. Region-based CNN models are the most popular algorithms to use for performing computer vision tasks. This algorithm was developed by Ross Girshick et al. He used the selective search approach to deduce 2000 regions from the image for classification. Faster R-CNN is considered a more advanced version of the R-CNN model, in terms of better accuracy and speed, thus it is a model that is being considered for use in our study [3].

Fig 1 shows the comparison in the time for, training. Figure 2 shows the comparison in the time for, testing. Thanh et al conducted a study that demonstrated four object detection models to detect dental caries using a smartphone [19]. Table 1, from the experiments conducted by the authors in [19] depicts the comparison between the different models. Of those four models, Faster R-CNN had the highest accuracy of 87.4%, followed by YOLO V3, having an accuracy of 83.4%. YOLO V3 is another population-based model that is very much in use in several fields. Unlike other models, which use several convolutional layers, the YOLO model tends to use only one convolution network that predicts the bounding boxes and does the classification of these boxes. This, this model has orders of magnitude (45 frames per second) more than any other model [16]. Both of these models, Faster R-CNN and YOLO V3 will be used to detect dental caries and at the end, the models will be evaluated using some performance metrics. The model that has better performance overall on all the metrics will be considered as the model to integrate within the hardware component. The hardware component looks to integrate a camera probe within a device that takes images of the frontal teeth, as well as, goes inside the mouth and takes intraoral images of the possible teeth from all angles.




(a)




(b)

**Fig.1.** Graphical illustration of (a) training time for different deep learning models (b) testing time for different deep learning models

| Model Deep Learning | Sensitivity % | Specificity % | Accuracy % | Precision % |
|---|---|---|---|---|
| YOLOv3 | 74 | 86.6 | 83.4 | 65.3 |
| Faster R-CNN | 71.2 | 92.9 | 87.4 | 77.3 |
| RetinaNet | 63.2 | 89.8 | 83 | 67.7 |
| SSD | 26 | 99.7 | 81 | 97.1 |

C: cavitated; NSC: no surface change; VNC: visually non-cavitated.

**Fig. 2.** Model evaluation of dental caries vs non-caries classification [19]

The four different models, shown above are explained below.

### 1.1 Faster R-CNN

This object detection model is divided into three parts:
1. Convolution layers, 2. Region Proposal Network (RPN), 3. Classes and Bounding Box prediction

The first part is the Convolutional Layers. It consists of two components: Base and Head. The base is crucial in retrieving the features while the head gets utilized in classifying the input data. The CNN model is comprised of several layers known as convolution and pooling layers as well as a fully connected layer, which serves the purpose of performing classification [10]. Feature extraction is divided into three steps Filtering, applying ReLu activation, and Max/Average Pooling. A filter gets applied to the input image of size n x m; n corresponds to the number of rows of the window whereas m corresponds to the number of columns for the window. This filter is better known as the kernel. The kernel often has a dimension of 3 x 3. Using this kernel, the convolution gets calculated by moving the kernel along the image that acts as the input and the result is a feature map. Feature maps contain the visual features that the kernel extracts. A ReLu activation function gets applied to this feature map, where all the negative values get promoted to zero and all the positive values remain the same. ReLu activation introduces non-linearity into the model and the training process gets completed at a faster rate. Finally, pooling enhances the visual features of that image and reduces the feature's dimension, hence reducing the number of computations to be performed. Max pooling will use a window to go over the feature map and extract the largest value from that window. The aim is to only keep the important features while discarding the unimportant ones.

Convolution layers serve the purpose of training filters to retrieve the significant features from the image, based on the object present in the image. Features could include texture, shape, edges, etc. For example, if one wishes to retrieve the important features of an animal, then the filters will be trained accordingly to learn the shapes, colors, and other features of that animal. The second component is known as the region proposal network. It is defined as a neural network that verifies whether there is an object or not and if there is, a bounding box gets predicted for that object. Finally, a fully connected neural network classifies the object class and draws a bounding box using a Support Vector Machine classifier [4].

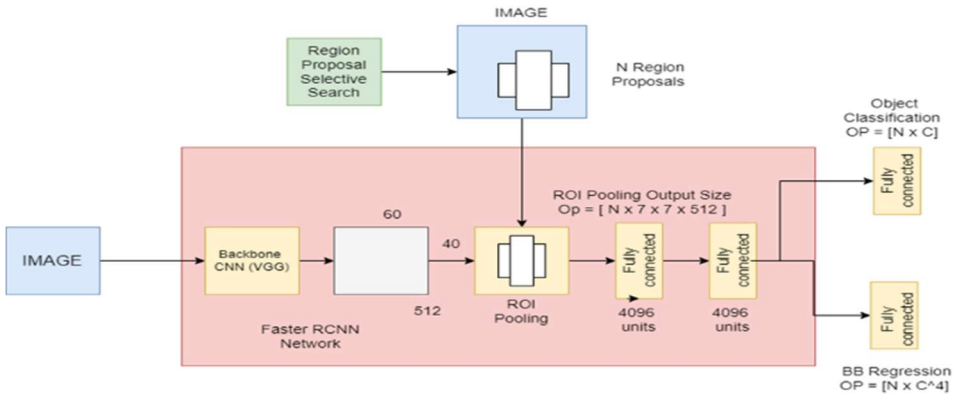



**Fig. 3.** Architecture of Faster R-CNN network using a backbone of Visual Geometry Group (VGG) with 16 layers

## 1.2 YOLO V3

It is an object detection model which only has a 1x1 convolution. A 1x1 convolution refers to the final prediction map having the same dimensions as the feature map. The YOLO V3 architecture is illustrated in Fig 4. The model gets an input image in the form of an array that consists of pixel values corresponding to the image [5]. The convolution network looks to recognize patterns in the image. This model "scores" region is based on their similarities to the already trained data. Regions with a high score are marked as positive detections with a class that they most identify with. This algorithm works by separating the image into a grid. The grid cells will predict the number of bounding boxes around the object, which tend to have a high score value. A confidence value is assigned to each bounding box indicating the accuracy of the prediction [12].

Each bounding can only detect one object. The bounding box is created by combining the measurements of the different ground truth boxes belonging to training data to discover the most similar shapes and sizes. This algorithm is considered fast, because of the base it uses for feature extraction. The base is known as Darknet-53. This architecture is made up of 24 convolutional layers and 2 fully connected layers. The 20 convolutional layers are met by a pooling layer as well as a fully connected layer. This base has already been trained on an ImageNet dataset. The layers consist of 3 x 3 convolutional layers and 1 x 1 reduction layers. In order to train the model, two fully connected layers and four convolutional layers get utilized. The last layer is used in predicting the probability of each and the bounding box. ReLu activation is applied to each layer, while a linear activation gets applied to that last layer.

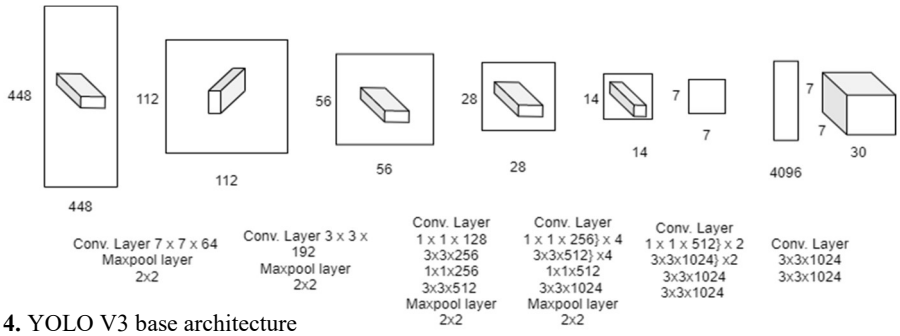



**Fig. 4.** YOLO V3 base architecture

### 1.3 Single Shot MultiBox Detector

Single Shot gets understood by the idea that the job of performing object classification and localization is done once during the forward propagation of the architecture. The technique to perform bounding box regression is called MultiBox. The detector is used to classify and locate the object. From Figure 5 the SSD architecture illustrates that the VGG-16 is used as the base for the convolutional neural network. VGG-16 is a high-performing network that is able to perform classification for high-quality images and is suitable for problems in which transfer learning is applicable. Some modifications were done to the VGG-16 base where a group of extra convolutional layers was added, allowing features to be retrieved at different scales. Through this process, the size of the input image decreases. The MultiBox loss function is a value indicating how well the model is behaving. The loss function is associated with two crucial parts: Confidence loss and Location loss. The confidence loss computes a value indicating how confident the network is in predicting an object within a bounding box. On the other hand, categorical cross-entropy calculates the loss. The location loss outputs a value giving a difference between the bounding boxes, predicted by the network, and the desired output boxes defined in the training data. L2-Norm gets utilized to compute loss. The logic behind drawing bounding boxes depends on MultiBox priors and Intersection over Union ratio.

Researchers developed a collection of pre-defined bounding boxes, of fixed size, known as priors. They had similarities in terms of distribution as compared to the ground truth boxes. Priors have to be selected, keeping the Intersection over Union (IoU) ratio in mind. Priors that have an IoU ratio larger than 0.5, have the bounding box, which got predicted, much closer to the ground or actual truth box [7]. The whole process behind training and running the SSD model is explained as follows. The first step is gathering data. Training and testing datasets would be required as well as bounding boxes that are referred to as the ground truth boxes and they get assigned labels for each class for the images. Some default bounding boxes have to be configured, each of different scales and sizes. Feature maps are produced to withdraw the visual attributes of the object in the image. Applying MultiBox on several feature maps allows an object of any size to be detected and localized. While training, most of the bounding boxes will have a low IoU ratio, and these samples are understood as negative training samples. The ratio of negative samples to positive samples is kept at 3:1. This will tell our network what is considered to be an incorrect detection. Data augmentation is performed so that the network becomes less prone to various object sizes. Additional training samples are created with the involvement of the original images at different IoU ratios. Finally, Non-Maximum Suppression gets performed which results in discarding bounding boxes with a confidence score of less than 0.01 and an IOU ratio of less than 0.45. Thus, the more likely predictions are kept while the noisier samples are eliminated.

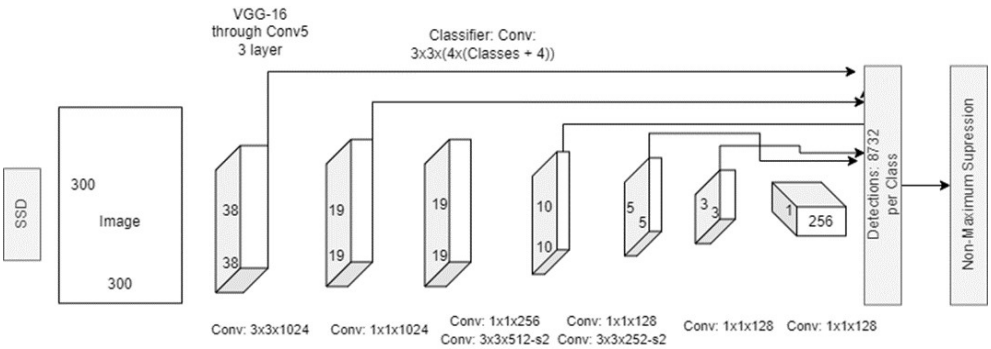



**Fig. 5.** Single Shot MultiBox Detector network architecture multiple convolutional layers

### 1.4 RetinaNet

RetinaNet got introduced by the Meta Artificial Intelligence research team for small object detection problems. The structure of the RetinaNet is composed of 3 parts:
1. Backbone Network
2. Sub-Network performing object classification
3. Sub-Network performing object regression[20]
The backbone network is further divided into two components:
The bottom-up pathway and the Top-down pathway with lateral connections. The bottom-up pathway gets used to develop feature maps at various dimensions, unbothered by the image size. The selection of spatially sparse feature maps that correspond to higher pyramid levels occurs via a top-down approach with lateral linkages. The top-down layers and bottom-up layers with comparable spatial sizes are connected by lateral connections. The second part of RetinaNet architecture, the Sub-Network for object classification, focuses on attaching a fully convolutional network to each Feature Pyramid Network for object classification. The sub-network consists of 3x3 convolutional layers with 256 filters. Subsequently, another convolutional layer of 3 x3 is attached with K * A filters attached. Thus, the feature map will have a size of W*H*KA. Sigmoid activation gets used for object classification at last.

The last component of RetinaNet architecture, the Sub-Network for object regression, involves attaching a regression network to the Feature Pyramid Network. The regression network has a similar design as the classification subnet. There lies a difference in the last convolutional layer, wherein the regression subnetwork, the size of this layer is 3 x 3 with four filters outputting a feature map with size W*H*4A. The regression subnetwork creates four amounts for each anchor box that estimates the difference regarding the width, height, and center of coordinates, between the anchor box and the desired box. 4A filters are included in the feature map. The base architecture is depicted in Figure 6. There are several parts depicted in the architecture. The four parts are ResNet, Feature Pyramid Net, class subnet, and box subnet. There are four major components in architecture.
1) Bottom-up pathway: Acts as the supporting backbone network which computes feature maps at multiple scales
2) Top-down pathway: This unsamples the spaced-out coarser feature maps from the pyramids at higher levels
3) Classification subnetwork: Making a prediction regarding the presence of an object in a spatial location
4) Regression Subnetwork: For each ground-truth item, it reverts the distance for the bounding boxes from the anchor boxes.

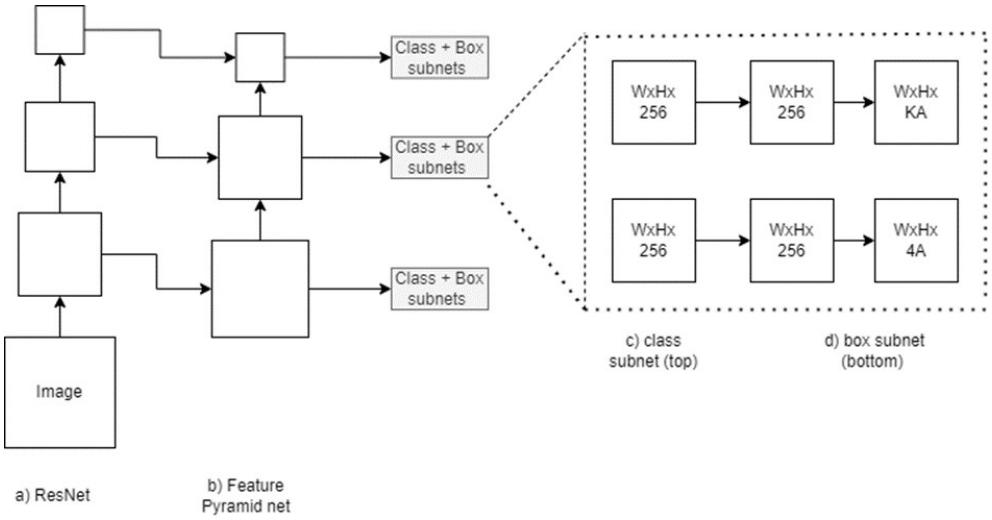



**Fig. 6.** RetinaNet architecture

## 2 Literature Review

Thanh et al proposed a mobile-based diagnostic application to detect dental caries anytime anywhere. They considered 4 deep learning models, which are Faster R-CNNs, YOLO v3, SSD, and RetinaNet. The models got trained on a total of 1902 images. Three angles were considered for the image to be taken: central view, left lateral view, and right lateral view. These images were classified into three different classes. Those classes indicated the state of the teeth. Of all the four models, Faster R-CNN and YOLOv3 had the best performance [19]. Patil et al offer an analysis of their algorithm for detecting dental caries through an adaptive neural network. The dental caries were detected through image pre-processing, feature extraction, and classification. The model used for classification was MPCA-ADA. Image pre-processing allowed the image to be enhanced in terms of pixel intensity, along with obtaining the region of interest through Otsu's thresholding and active contours. Multilinear Principal Component Analysis (MPCA) helps in extracting features and gets multiplied by the weight that gets used as an input to the feed-forward network for performing classification. Nonlinear programming optimization was introduced to maximize the distance between the selected features, thus improving the classification performance. The authors ran three test cases and found that MPCA-ADA was roughly 20% more accurate than all other algorithms (PCA-ADA, LDA-ADA, ICA-ADA), for each test case [15].

Kühnisch et al detected caries on intraoral images using convolutional neural networks. Many of the images were filtered out due to the quality. CNN was trained on a total of 2417 images. The model was trained through a pipeline of numerous predetermined functions such as transfer learning and image augmentation. Image augmentation provided a large number of images to the model continually. To speed up the training procedure, a neural network along with predefined weights get utilized, named MobileNetV2. The CNN has an accuracy of 92.5% after considering 100% of the images from the training data. Furthermore, the results can be ameliorated by increasing the quantity of images for training and, thus improving image segmentation [8].

Eggert et al provided an in-detail review regarding the detection of small objects using Faster R-CNN. They applied this algorithm to the detection of the company logo. Since company logos were considered small objects, they theoretically examined the problem of small objects and drove a relationship that described the minimum object size. They carried out experiments that used features from multiple feature maps to understand how the proposal and classification phases behaved in relation to object size. They evaluated the results in the Flickr Logos Dataset [4].

The selective-search algorithm, used in Faster R-CNN, gets utilized to select 2000 region proposals from an image in a top-down manner [11].

The neural network of YOLO is made up of several components that include candidate box extraction, classification of objects, and feature extraction [9].

Most of the classifiers used in CNN are SVM, and its purpose is to combine all the results from the feature extraction phase [18].

The benefit of CNN is that there are a minimum number of parameters as compared to a fully connected neural network because the receptive attributes in the layer share weights [14].

Average Object Area Recall is a performance metric used to evaluate object detection models. It is defined as the mean area recall of the actual objects stored in the data. The recall is the portion of the area that gets covered by the result boxes of the algorithm [13].

## 3 Methodology

Two methodologies have been proposed to implement the following problem of detecting dental caries. The dataset comprised roughly of 300 images, where roughly 250 images acted as the training data and 50 images acted as the testing data. The following two methods are proposed below.

### 3.1 YOLO V3

It is an object detection model that is considered an optimized algorithm for object detection and classification. The notion of this model is that the class probabilities and bounding boxes will be predicted for the image through a single neural network in one evaluation. A single network is consisting of a whole detection pipeline; thus, it can be optimized end-to-end on the parameter of detection performance.

Several techniques have been employed to detect objects; however, they make use of a pipeline execution architecture, which forces the network on every individual component separately. This leads to increased time for training and complexity for optimization. The YOLO V3 algorithm takes an input image from the camera and forwards it to the neural network that produces an output vector defining the coordinates for the bounding box and class probabilities. This algorithm makes use of a 53-layer network that is trained on Imagenet known as Darknet-53 as shown in Figure 7. This is the feature extractor for the architecture. While doing detection, 53 more layers are placed on top of the framework to give us a 106-layer network supporting the architecture.The YOLO V3 algorithms functions by splitting the input image into N grid cells of size M x N. Each grid can perform detection and localization of the image.

Subsequently, all the grid cells are able to predict the bounding box coordinates for the object along with the class probabilities and class labels. Detection is achieved by applying a kernel of size 1x1 on the feature map consisting of different sizes at multiple locations in the structure. The detection kernel has the following dimensions 1x1x(Bx(5+C)). B equates to the quantity of bounding boxes the cells on the feature map can estimate. The '5' comprises of four bounding box features and one object confidence. Lastly, 'C' refers to the amount of

classes. Binary cross-entropy aids in calculating loss for classification while logistic regression is made used for estimating object probability and class probabilities. As mentioned above, the YOLO v3 tends to take an input image and produce an output vector.

The output vector consists of the following parameters:

1) Class probabilities: This defines that the probability of identifying an object in the bounding box is associated with some specific class.

2) Bounding box values: The bounding boxes' height and width are given, as well as the cartesian position of that box.

3) Prediction probability: The several bounding boxes containing a detectable object are represented as a probability.

The YOLO V3 model was used for object detection through the following steps:

I. Data acquisition: Roughly 300 images of teeth were acquired of which 80% was used for training and 20% was used for testing purposes.

II. Data Labelling: The LabelImg tool is used to label each and every image, as well drawing the ground truth box for each of the caries in the image. At the end of the process, a text file got generated for the entire dataset. The file consists of information regarding the image id and the coordinates of the bounding box.

III. Feature Extraction: Using the darknet-53 framework, the key features of the images were extracted and the model was trained. The estimated time for training was 9 hours and the number of iterations taken was above 2000. At the end of the training, two files get generated by the name "yolov3_training_last.weights" and another file "yolov3_testing.cfg". These files are the primary component for performing real-time detection.

IV. Testing Object detector: The files generated in the feature extraction phase get used, along with the OpenCV library, and real-time object detection gets performed on the testing images.

V. Anchor Boxes

There are cases where the midpoints of several objects fall on the same grid cell, making it tedious to detect those objects. To overcome this problem, each object in the same grid is associated with an anchor box. For example, if there are two anchor boxes associated with the two objects, then there would two predictions in the same grid. The Intersection over Union ratio is calculated, corresponding to that object. If the value obtained is less than the threshold value (let's say 0.5), then that object won't be considered for detection.

VI. Non-Maximum Suppression

The last step in YOLO v3 involves solving an issue that arises when multiple bounding boxes have been detected for the same object. The bounding boxes tend to overlap each other and non-maximum suppression has the job of identifying the best bounding box of all of them. The IoU is calculated for each bounding box and compared with the threshold. If any bounding box has a lower IoU than the threshold, then it gets eliminated. If all the bounding boxes have an IoU ratio greater than the threshold, then the bounding box with the maximum IoU ratio is considered.

Several performance metrics have been mentioned in order to measure the working of the object detection model, as done by the author Zhao et al. [21].

1) Precision

Precision is defined as the proportion of the number of objects that are detected correctly to the number of total objects detected overall. It is represented by:

$$Precision = \frac{TP}{TP+FP} \quad (1)$$

TP corresponds to True Positive and FP means False Positive.

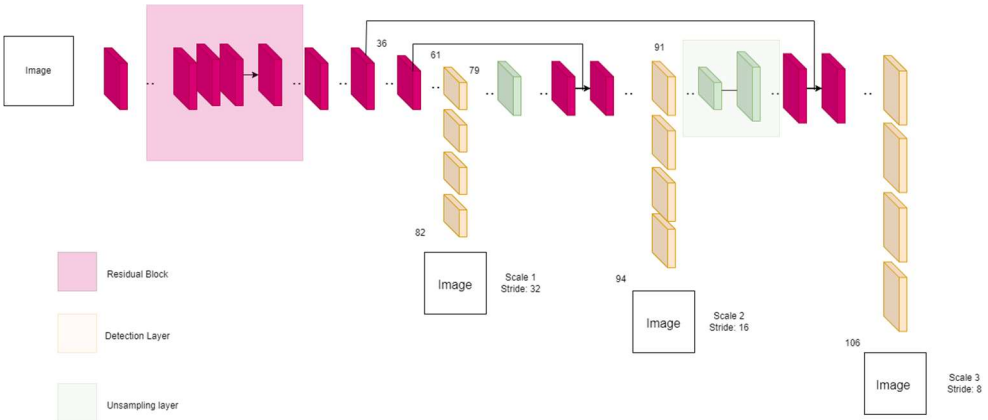



**Fig. 7.** Darknet-53 architecture used in YOLO V3 as the backbone for feature extraction

2) Specificity
It describes the number of true negatives identified correctly. This insinuates that more true negatives, that were once thought to be positive and could correspond to false positives, will be noted. High specificity refers to the model correctly identifying the negative results. It is given by the following formula:

$$Specificity = \frac{TN}{TN+FP} \quad (2)$$

Where TN is equivalent to True Negative

3) Sensitivity
Sensitivity defines how well the model can correctly predict the positive test cases. It evaluates the model's performance as it gives an idea of the number of positive instances that were identified correctly. The higher the sensitivity, the more correctly it is able to predict the positive instances.

$$Sensitivity = \frac{TP}{TP+FN} \quad (3)$$

Where FN means False Negative.
4)Accuracy
It is the ratio of objects with accurate labels to the total set of objects.

$$Accuracy = \frac{TP+TN}{TP+TN+FN+F} \quad (4)$$

5) Intersection over Union Ratio
The degree to which the anticipated bounding box comes across the ground truth box is how it is described. IoU is a performance metric, and Equation (5) represents it, as seen in Figure 8. Here A and B are referencing the bounding box for the prediction and the ground truth box respectively.

$$IoU = \frac{(A \cap B)}{(A \cup B)} \quad (5)$$

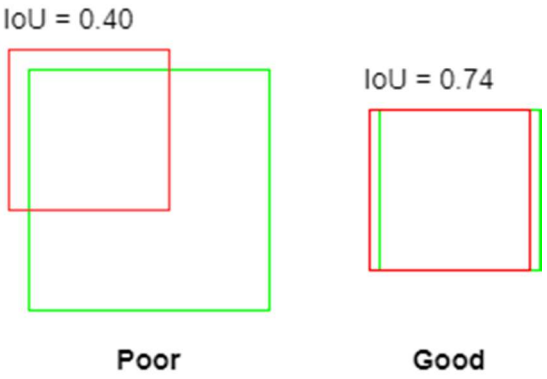



**Fig.8.** Depiction of Intersection over Union Ratio

### 3.2 Faster R-CNN

It is seen as an advancement to Fast R-CNN. The architecture can be seen in figure 9. The algorithm can be briefly explained in the following steps:
1. Region Proposal Network (RPN) forms region proposals
2. The Region of Interest (ROI) Pooling layer helps extract a feature vector of all the regions corresponding to the region proposals in the image.
3. Using the structure of Fast R-CNN, the feature vector extracted in the previous step is classified.
4. The bounding box coordinates and the class scores are estimated.

#### *3.2.1 Convolution Layers*

Firstly, the image acts as an input to the convolution layers to generate feature maps. The ensuing RPN layers share the feature maps. RPN is the resultant feature map that gets generated from the last convolutional layer. The Fast R-convolutional CNN layers are employed by the RPN to process the image. As a result, creating the proposals is quicker with the RPN than with a selective-search algorithm. A window size of n x n will be passed across the feature map to generate region proposals. These region proposals will be filtered based on the objectness score. A VGG-16 backbone gets used in the convolution layers which consist of a total of 13 convolution layers, along with the ReLu activation function, to introduce non-linearity and 4 pooling layers. The convolution layers are able to retain the original size of the input image throughout processing.

#### *3.2.2 Region proposal Network*

The RPN produces a group of proposals. Those proposals have values associated with them. First, is the score telling the chance of being an object. Second, the class label identifies the object. Proposals get generated by the mechanism of sliding a small window of size n x n across the feature map. A low-dimensional feature is associated with each sliding window. The position of the sliding window provides information about the image's localization.

#### *3.2.3 Anchor Boxes*

An important concept that is responsible for acting as a predefined bounding box in reference to when first making a prediction of the object location. Anchor boxes get used in order to evaluate the entire object predictions at once. They aid in improving the efficiency and time

for the detection task in this algorithm. They are also able to detect multiple objects having different scales.

### *3.2.4 ROI Pooling*

Input feature maps and proposals are stored through the ROI pooling layer. It helps in time. After performing pooling, a small feature map of fixed size gets produced. The feature map is of size 7 x 7. This emphasizes the most important aspects of the image. The purpose of this is to reduce training

### *3.2.5 Classification*

The proposal feature maps are considered significant to compute the proposal's label and coordinates for the bounding box. The classifier gets a feature map of size 7 x 7 as input. Using the SoftMax activation function, the classifier correlates which proposal belongs to which class, for example, caries, or non-caries. This results in an output vector getting produced.

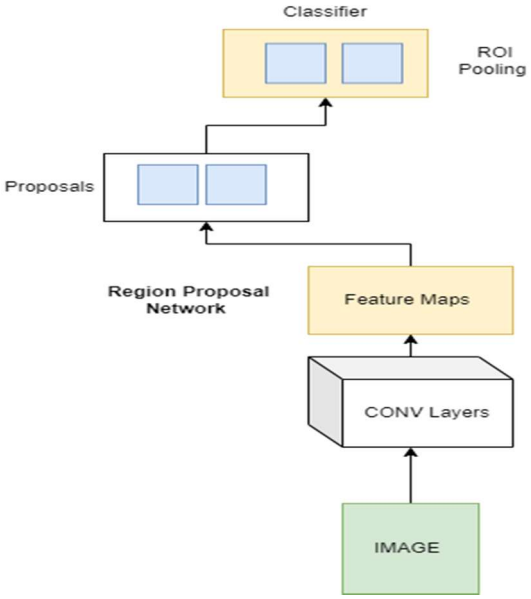



**Fig.9.** Faster R-CNN block diagram

## 4 Results and Discussion

Table 2 depicts the results obtained from performing object detection using the two algorithms: Faster R-CNN and YOLO V3. Faster R-CNN's accuracy was recorded at 80%, slightly higher than the YOLO V3 algorithm's accuracy of 75%. The sensitivity of the YOLO v3 algorithm was 76%, whereas the sensitivity of the Faster R-CNN was recorded as 73%. The precision of Faster R-CNN was higher, 78%, as compared to that of YOLO v3, 74%. Most of the performance metrics favoured the Faster R-CNN algorithm. Figure 10 illustrated the live working of the Faster R-CNN algorithm to locate dental caries on one of the testing images.

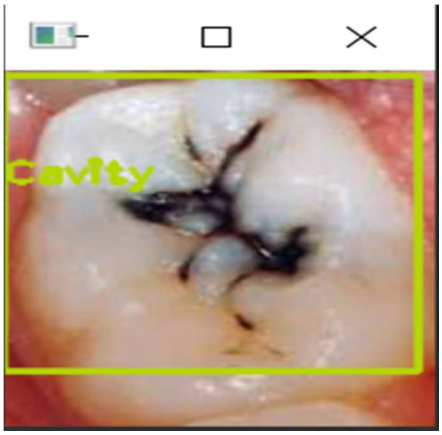



**Fig. 10.** Dental caries detection using YOLO V3

**Table 1.** Comparison between the two proposed models using four different performance metrics

| **Methods** | **Accuracy** | **Specificity** | **Sensitivity** | **Precision** |
|---|---|---|---|---|
| **YOLO V3** | 75% | 72% | 76% | 74% |
| **Faster R-CNN** | 80% | 78% | 73% | 78% |

Our two models were trained on a total dataset of 300 images. This allowed the model to learn the key features of images that consisted of dental caries. Caries is clear on any teeth by differentiating them from regular white teeth. The testing images consisted of both non-caries teeth and teeth with caries. Different angles of the teeth were taken to ensure the orientation is not an issue during real-time testing. From the experiments conducted, Faster R-CNN was the more dominant one in terms of performance. It provides a higher accuracy, 80%, than YOLO V3, 75%, even though both were trained on the same number of images. The model performed well for a sufficiently low dataset. Based on the results of the object detection models, our aim is to use the better-performing one as the software component for the device. The hardware looks to contain a high-definition camera probe that goes inside the mouth. The intraoral images will be taken and transferred to our software module. Then the object detection model will look to detect dental caries if there happens to be one. The difference in performance between YOLO v3 and Faster R-CNN is quite less, however, it is sufficient enough to declare which model to use for our system. Faster R-CNN is the most advanced algorithm as compared to previous Region based CNNs. While previous algorithms made use of Selective search, Faster R-CNN looks to use Region Proposal Network. It has a strong base, VGG-16, which is a pre-trained algorithm that extracts features from the given image and produces a feature map. VGG-16, being a strong base is the key success to its

great performance. On the contrary, YOLO v3 was able to evaluate the image on the first go, so though it was faster as there are many repetitions, the performance was sacrificed.

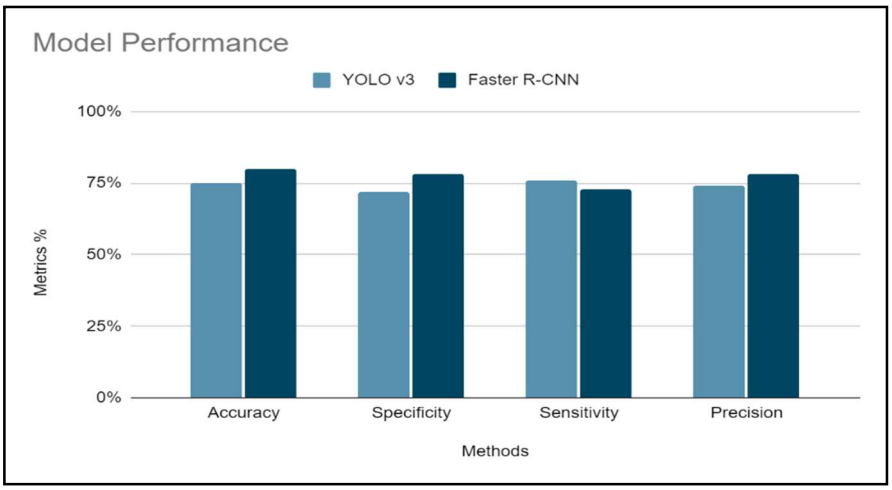



**Fig. 11.** Bar graph showing the comparison between the two models with respect to the four-performance metrics

## 5 Conclusion

Two algorithms were implemented to detect dental caries: Faster R-CNN and YOLO v3. The dataset consisted of roughly 300 images. 250 images were used in the training set and 50 images were used in the testing set. It is observed from table 2 that Faster R-CNN performed better than the YOLO v3 algorithm in most of the parameters. In comparison to the authors who performed similar experiments in [19], the accuracy of their algorithms was roughly similar. Figure 11 illustrates a bar graph showing the comparison of the two proposed models. The accuracy they achieved for the YOLO v3 algorithm was roughly 83%, whereas our algorithm received an accuracy of 75%. Similarly, they achieved an accuracy of 87% for the Faster R-CNN model. In contrast, our Faster R-CNN algorithm achieved an accuracy of 80%. In the future, datasets could have been increased from various dental hospitals in order to increase the range of data. Perhaps, data could go up to 1000 images. In the future, a larger variety of datasets should be used to improve our results. Our study focuses on training machine algorithms to differentiate between the different levels of dental caries present in an individual. While our team looked at two well-trained algorithms, for the future scope, there could be an exploration regarding several algorithms such as Retinanet, Single Shot Detector, etc.

## References

1. Cao, C., Wang, B., Zhang, W., Zeng, X., Yan, X., Feng, Z., ... & Wu, Z. (2019). An improved faster R-CNN for small object detection. *Ieee Access*, *7*, 106838-106846.
2. Casalegno, F., Newton, T., Daher, R., Abdelaziz, M., Lodi-Rizzini, A., Schürmann, F., ... & Markram, H. (2019). Caries detection with near-infrared transillumination using deep learning. *Journal of dental research*, *98*(11), 1227-1233.
3. Datta, S., & Chaki, N. (2015, November). Detection of dental caries lesion at early stage based on image analysis technique. In *2015 IEEE International Conference on Computer Graphics, Vision and Information Security (CGVIS)* (pp. 89-93). IEEE.
4. Eggert, C., Brehm, S., Winschel, A., Zecha, D., & Lienhart, R. (2017, July). A closer look: Small object detection in faster R-CNN. In *2017 IEEE international conference on multimedia and expo (ICME)* (pp. 421-426).IEEE.
5. Gavrilescu, R., Zet, C., Foșalău, C., Skoczylas, M., & Cotovanu, D. (2018, October). Faster R-CNN: an approach to real-time object detection. In *2018 International Conference and Exposition on Electrical And Power Engineering (EPE)* (pp. 0165-0168). IEEE.
6. Huang, R., Pedoeem, J., & Chen, C. (2018, December). YOLO-LITE: a real-time object detection algorithm optimized for non-GPU computers. In *2018 IEEE International Conference on Big Data (Big Data)* (pp. 2503-2510). IEEE.
7. Kanimozhi, S., Gayathri, G., & Mala, T. (2019, February). Multiple Real-time object identification using Single shot Multi-Box detection. In *2019 International Conference on Computational Intelligence in Data Science (ICCIDS)* (pp. 1-5). IEEE.
8. Kühnisch, J., Meyer, O., Hesenius, M., Hickel, R., & Gruhn, V. (2022). Caries detection on intraoral images using artificial intelligence. *Journal of dental research*, *101*(2), 158-165.4
9. Lee, J. H., Kim, D. H., Jeong, S. N., & Choi, S. H. (2018). Detection and diagnosis of dental caries using a deep learning-based convolutional neural network algorithm. *Journal of dentistry*, *77*, 106-111.
10. Lian, L., Zhu, T., Zhu, F., & Zhu, H. (2021). Deep learning for caries detection and classification. *Diagnostics*, *11*(9), 1672.
11. Liu, B., Zhao, W., & Sun, Q. (2017, October). Study of object detection based on Faster R-CNN. In *2017 Chinese Automation Congress (CAC)* (pp. 6233-6236). IEEE.
12. Liu, C., Tao, Y., Liang, J., Li, K., & Chen, Y. (2018, December). Object detection based on YOLO network. In *2018 IEEE 4th Information Technology and Mechatronics Engineering Conference (ITOEC)* (pp. 799-803). IEEE.
13. Mariano, V. Y., Min, J., Park, J. H., Kasturi, R., Mihalcik, D., Li, H., ... & Drayer, T. (2002, August). Performance evaluation of object detection algorithms. In *2002 International Conference on Pattern Recognition* (Vol. 3, pp. 965-969). IEEE.
14. Minaee, S., Boykov, Y. Y., Porikli, F., Plaza, A. J., Kehtarnavaz, N., & Terzopoulos, D. (2021). Image segmentation using deep learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*
*15.* Patil, S., Kulkarni, V., & Bhise, A. (2019). Algorithmic analysis for dental caries detection using an adaptive neural network architecture. *Heliyon*, *5*(5), e01579.
16. Shafiee, M. J., Chywl, B., Li, F., & Wong, A. (2017). Fast YOLO: A fast you only look once system for real-time embedded object detection in video. *arXiv preprint arXiv:1709.05943*.
17. Srivastava, M. M., Kumar, P., Pradhan, L., & Varadarajan, S. (2017). Detection of tooth caries in bitewing radiographs using deep learning. *arXiv preprint arXiv:1711.07312*.
18. Thai, L. H., Hai, T. S., & Thuy, N. T. (2012). Image classification using support vector machine and artificial neural network. *International Journal of Information Technology and Computer Science*, *4*(5), 32-38.

19. Thanh, M. T. G., Van Toan, N., Ngoc, V. T. N., Tra, N. T., Giap, C. N., & Nguyen, D. M. (2022). Deep Learning Application in Dental Caries Detection Using Intraoral Photos Taken by Smartphones. *Applied Sciences*, *12*(11), 5504
20. Zhang, H., Chang, H., Ma, B., Shan, S., & Chen, X. (2019). Cascade retinanet: Maintaining consistency for single-stage object detection. *arXiv preprint arXiv:1907.06881*.
21. Zhao, L., & Li, S. (2020). Object detection algorithm based on improved YOLOv3. *Electronics*, *9*(3), 537.
22. Zhao, Z. Q., Zheng, P., Xu, S. T., & Wu, X. (2019). Object detection with deep learning: A review. *IEEE transactions on neural networks and learning systems*, *30*(11), 3212-3232.