COL780 Assignment 4 REPORT

Ananya Mathur - 2020CS50416
Thakre Ishita - 2020CS50445

April 2024

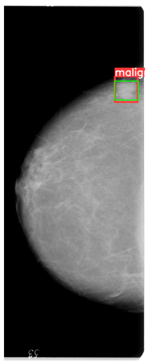
1 Convolution Based Models

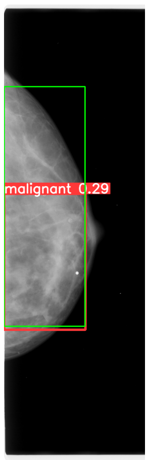
We tried out faster RCNN model and a YOLO based Model for this part.

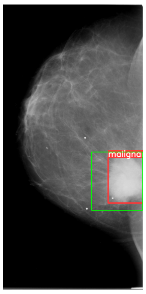
1.1 YOLO Based Model

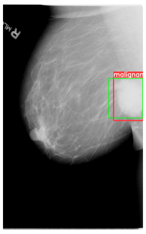
We have used the repository of YOLOv5 for this part. The confidence threshold used by us is 0.25 and the IOU threshold is 0.45. These were the default ones. We didn't change them.

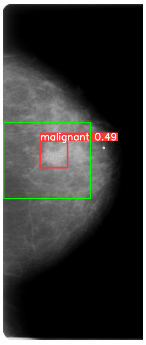
1.1.1 Data Visualisation

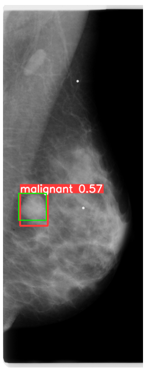


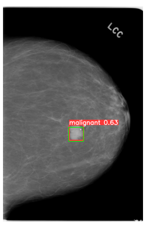


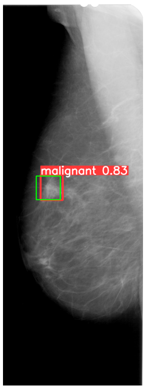


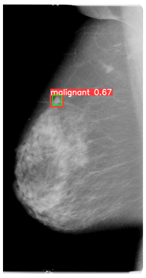



malignant 0.49


malignant 0.57







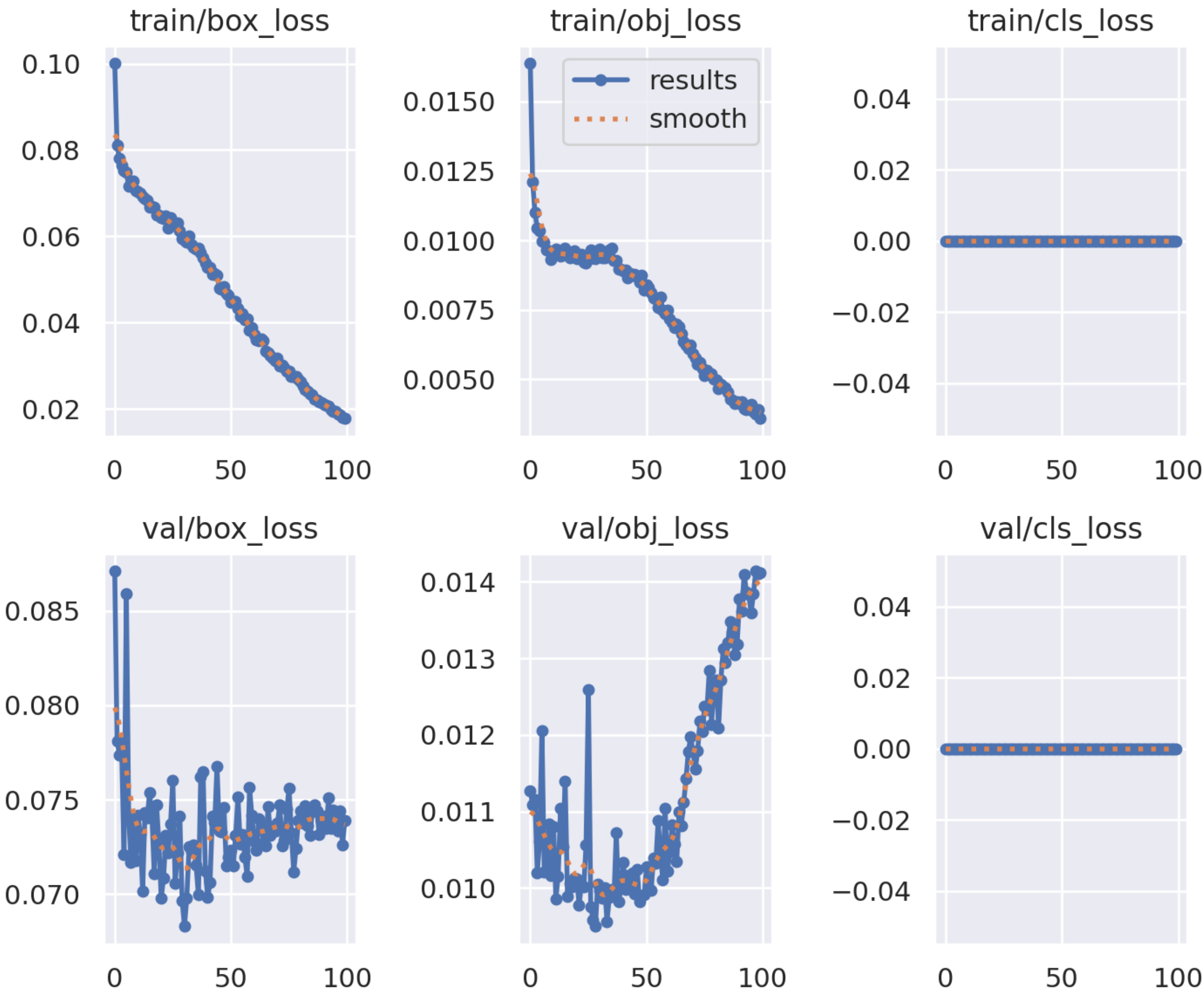
The above images show the predicted boxes and the actual boxes in the dataset.
There were some images on which the tumours went undetected, but there were fewer such images where boxes were wrongly detected. We assume that decreasing the confidence threshold would give us better results.

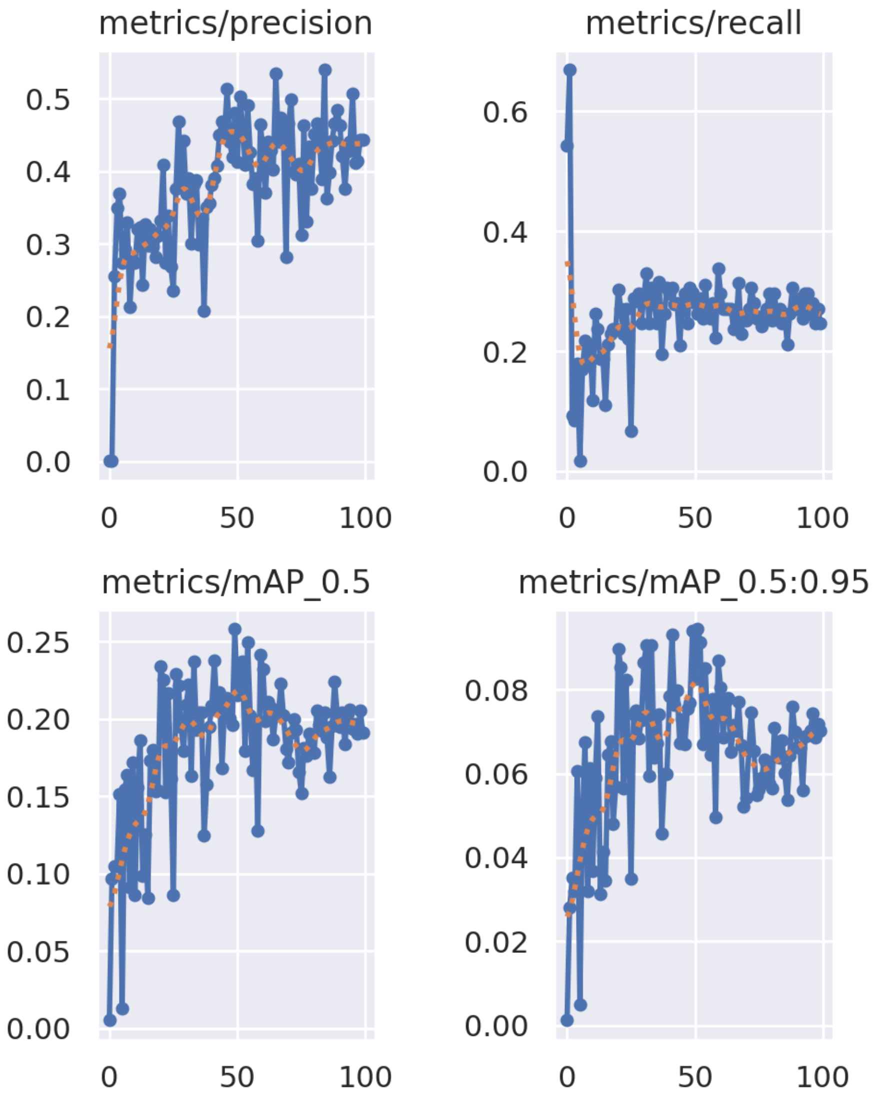
1.1.2 Image Preprocessing

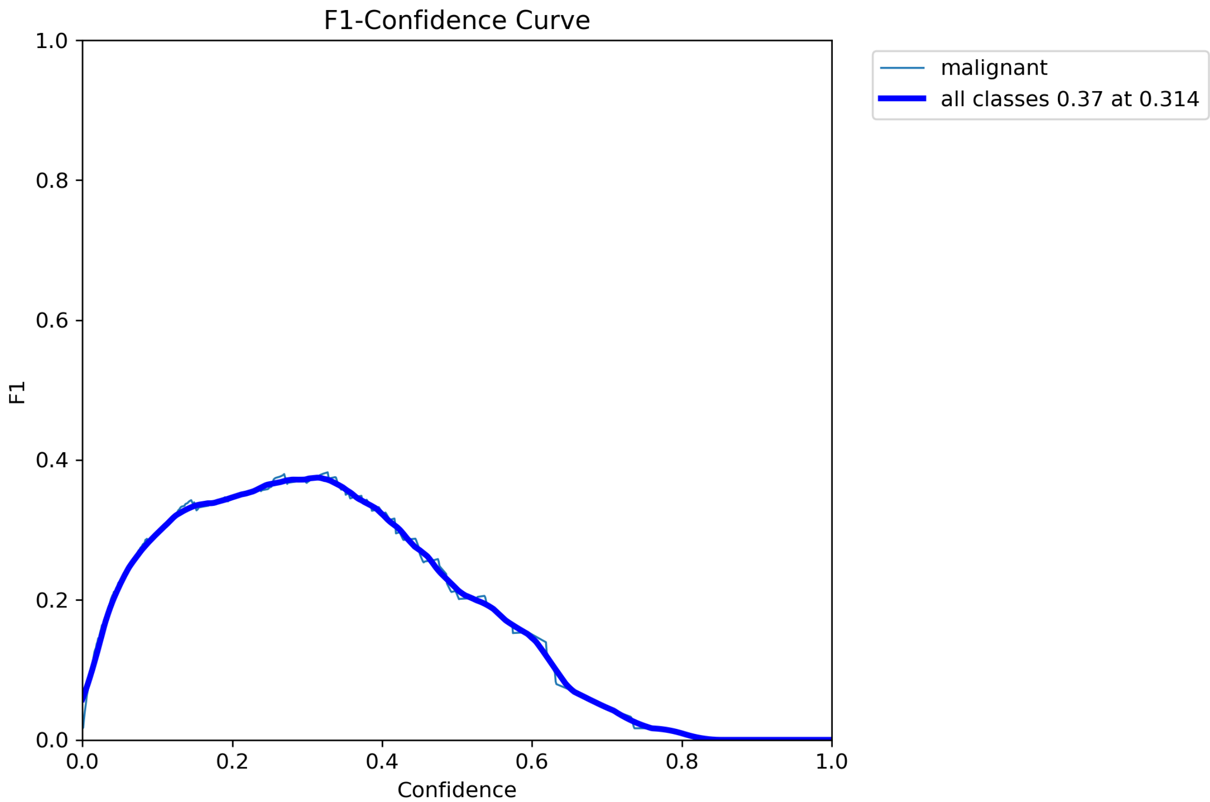
Resizing
YOLO models typically expect images of a certain size as input. It is required to resize the images to match the input dimensions required by the YOLO model. We resized images to 640 x 640. Resizing helps ensure that the model processes the entire image and captures objects at different scales. Many object detection models require input images to have a fixed size. Resizing ensures that all images fed into the model have the same dimensions, which is necessary for the model to process them efficiently and make predictions accurately. Resizing images to a smaller size reduces the memory and computational resources required to process them. It helps to train faster
Padding
We padded images with zero pixels to match the input dimensions of the YOLO model if necessary. Padding ensures that all images have the same dimensions, which is required by most deep learning models.

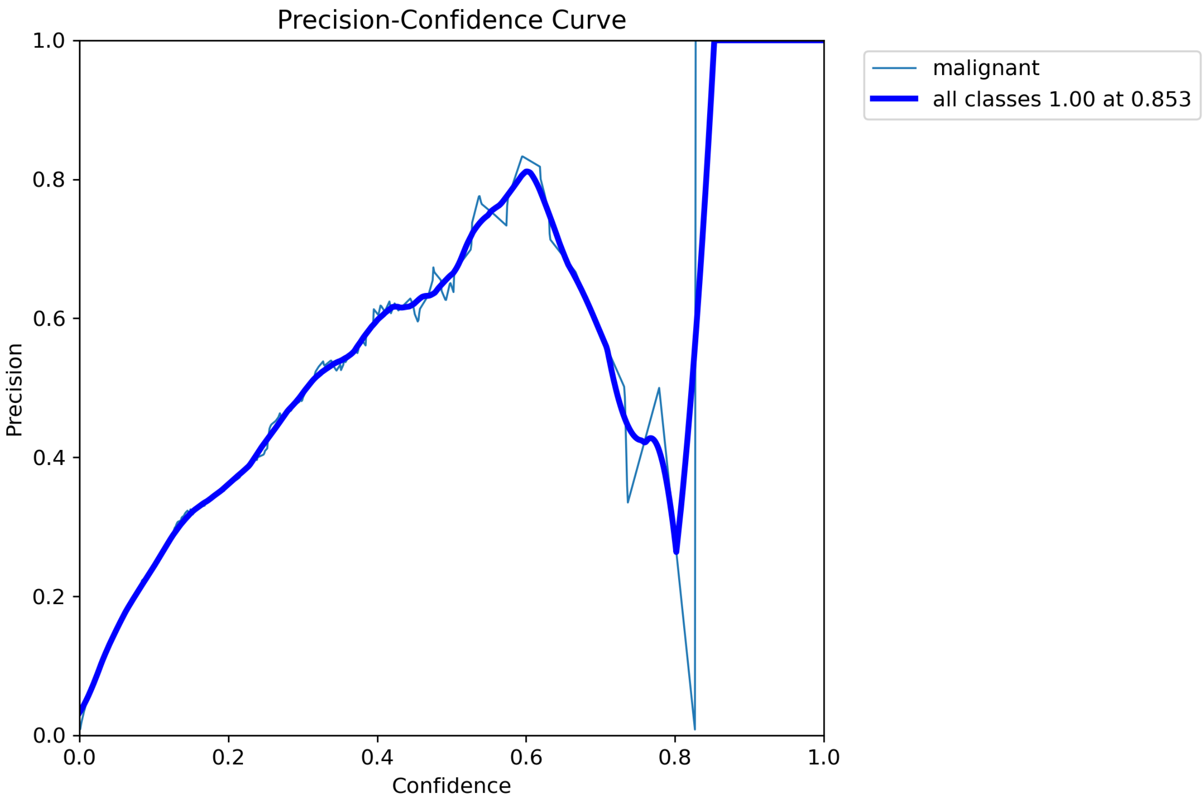
1.1.3 Training and validation loss curves

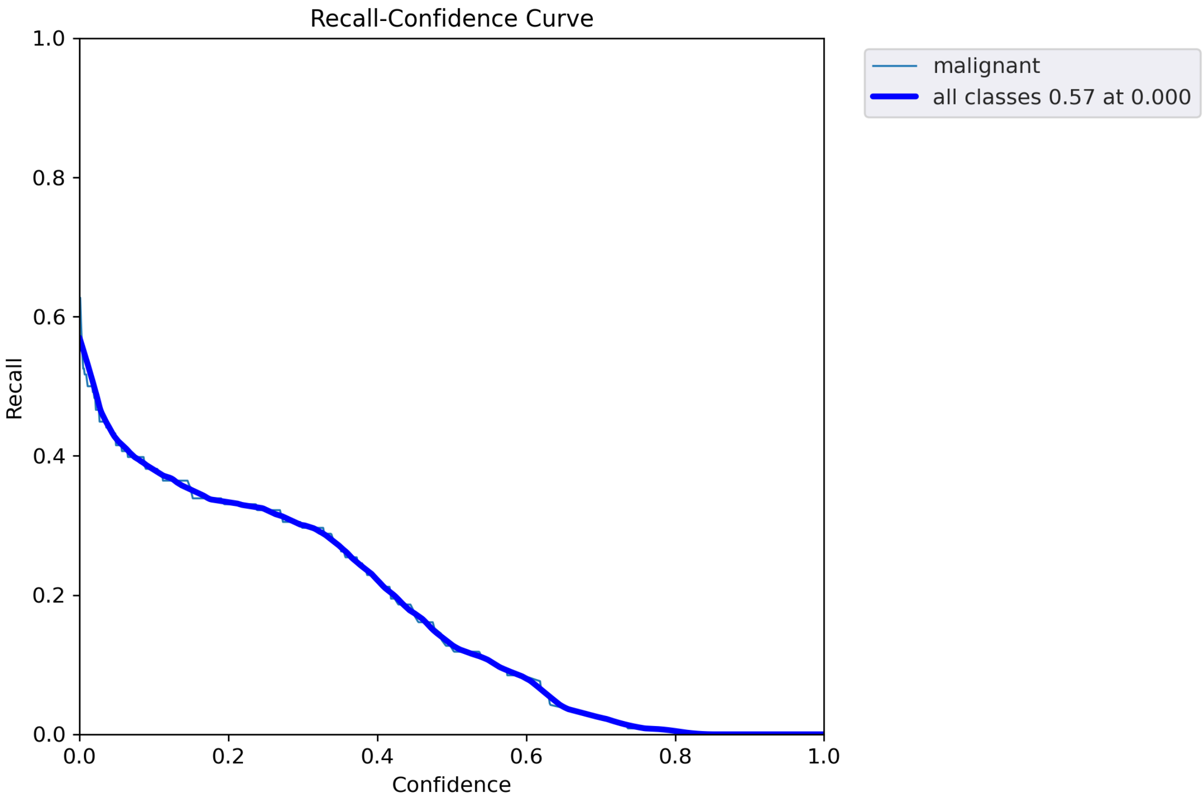
The training and validation loss curves are attached below

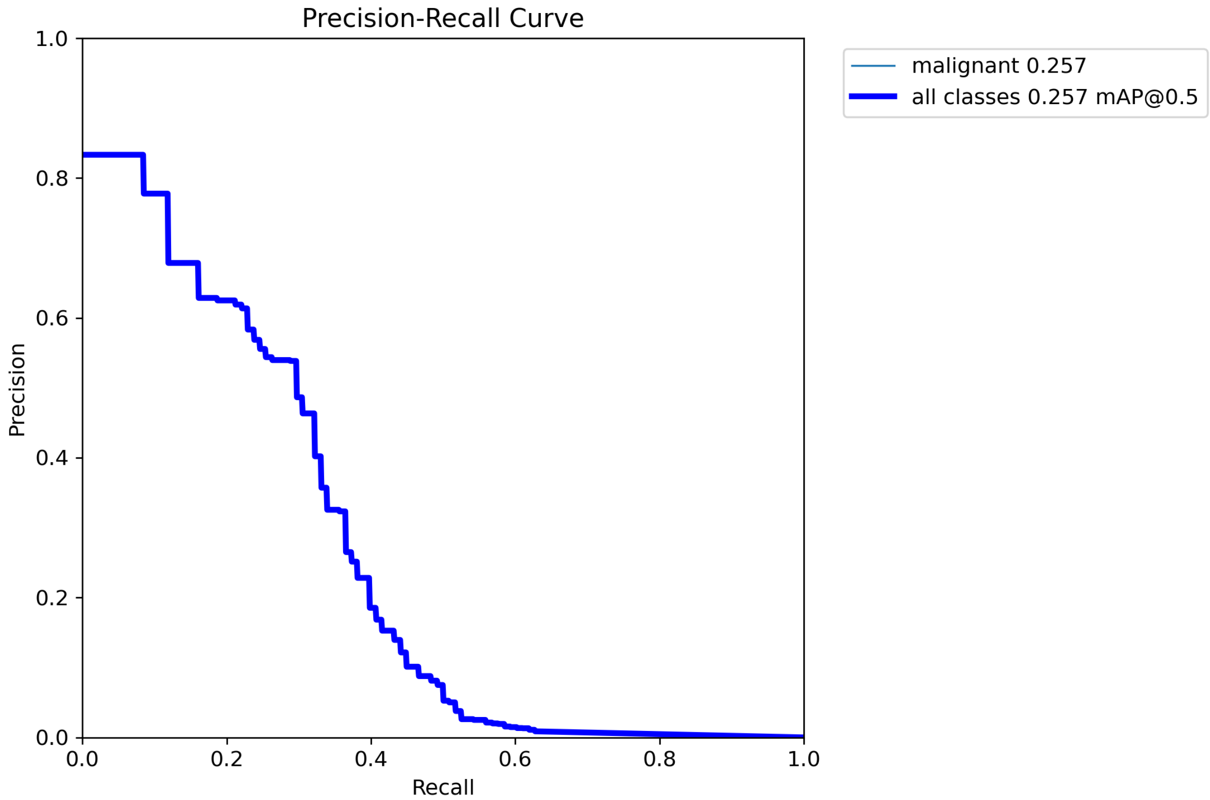

train/box_loss
train/obj_loss
train/cls_loss
val/box_loss
val/obj_loss
val/cls_loss
results
smooth


metrics/precision
metrics/recall
metrics/mAP_0.5
metrics/mAP_0.5:0.95

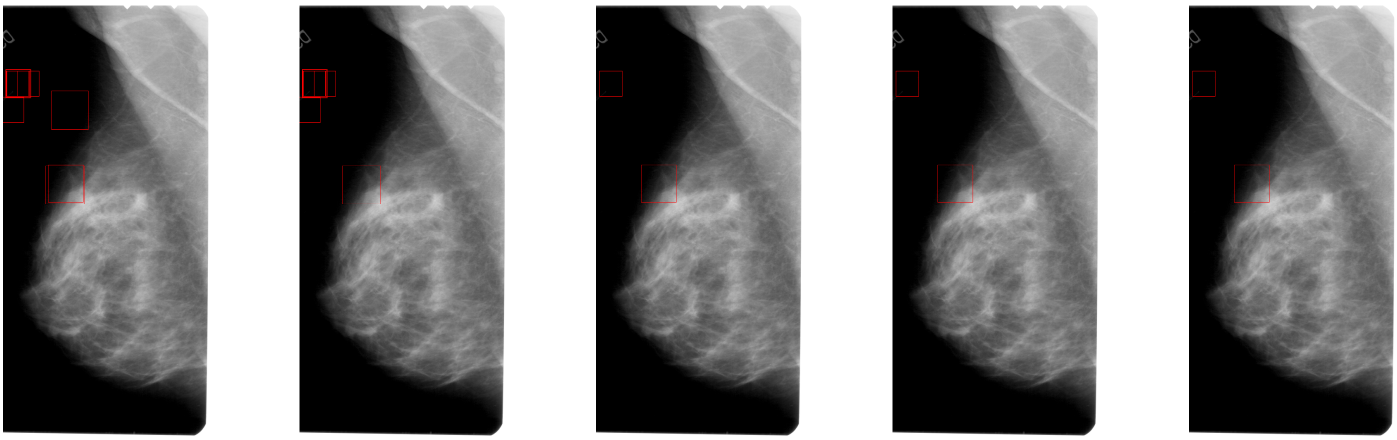

F1-Confidence Curve
F1
Confidence
malignant
all classes 0.37 at 0.314


Precision-Confidence Curve
malignant
all classes 1.00 at 0.853
Precision
Confidence


Recall-Confidence Curve
malignant
all classes 0.57 at 0.000
Recall
Confidence



1.1.4 Non-Maximum Suppression



(a) Threshold : 1 (9 malignants)

(b) Threshold : 0.5 (7 malignants)

(c) Threshold : 0.25 (2 malignants)

(d) Threshold : 0.1 (2 malignants)

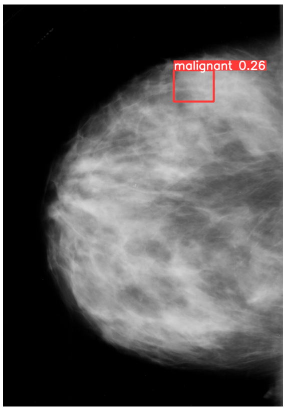
(e) Threshold : 0 (2 malignants)

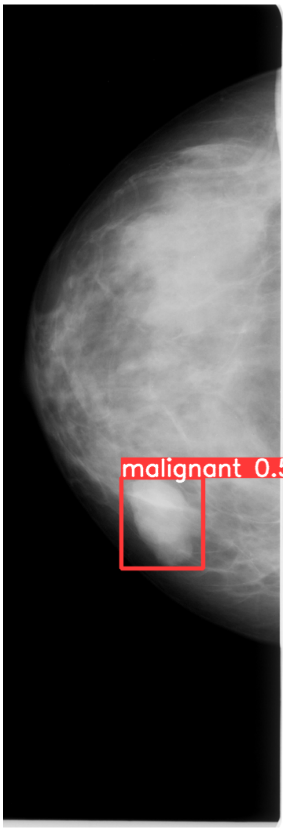
Figure 1: NMS

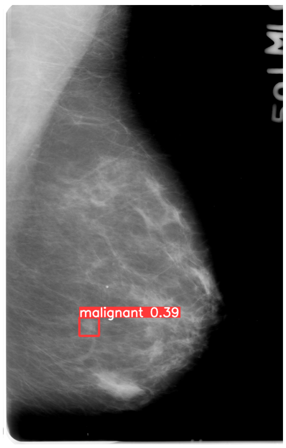
1.1.5 Grad-CAM/Attention Maps visualizations

1.2 faster RCNN Model

1.2.1 Data Visualisation







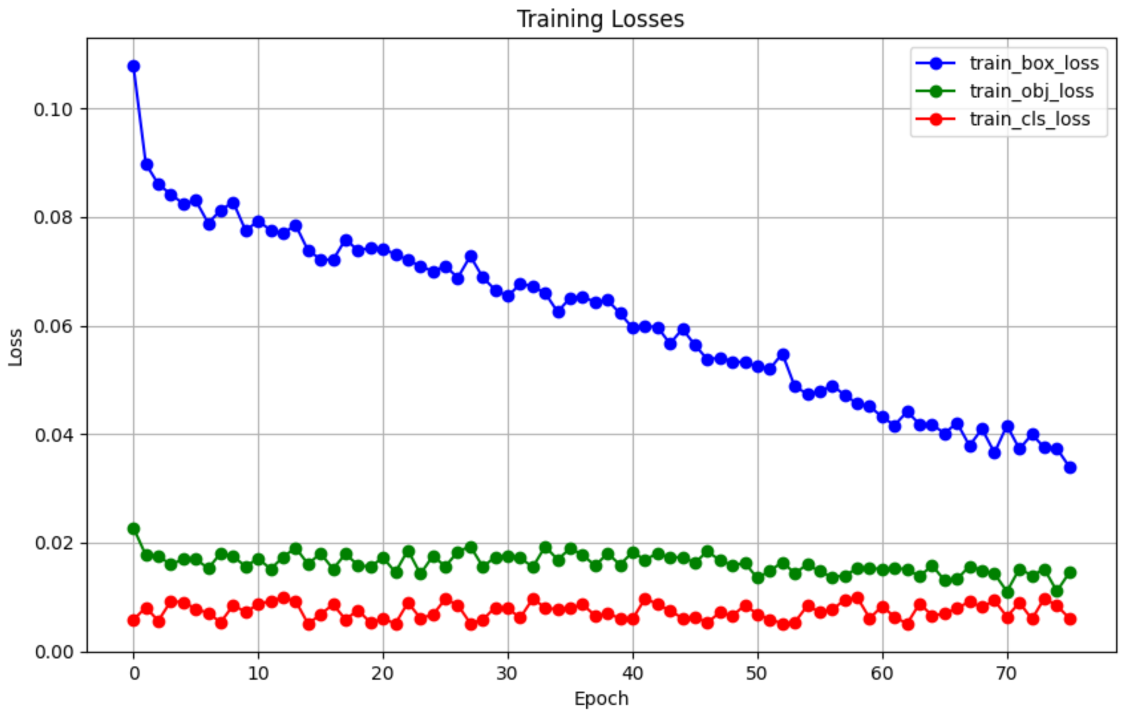
1.2.2 Image Preprocessing

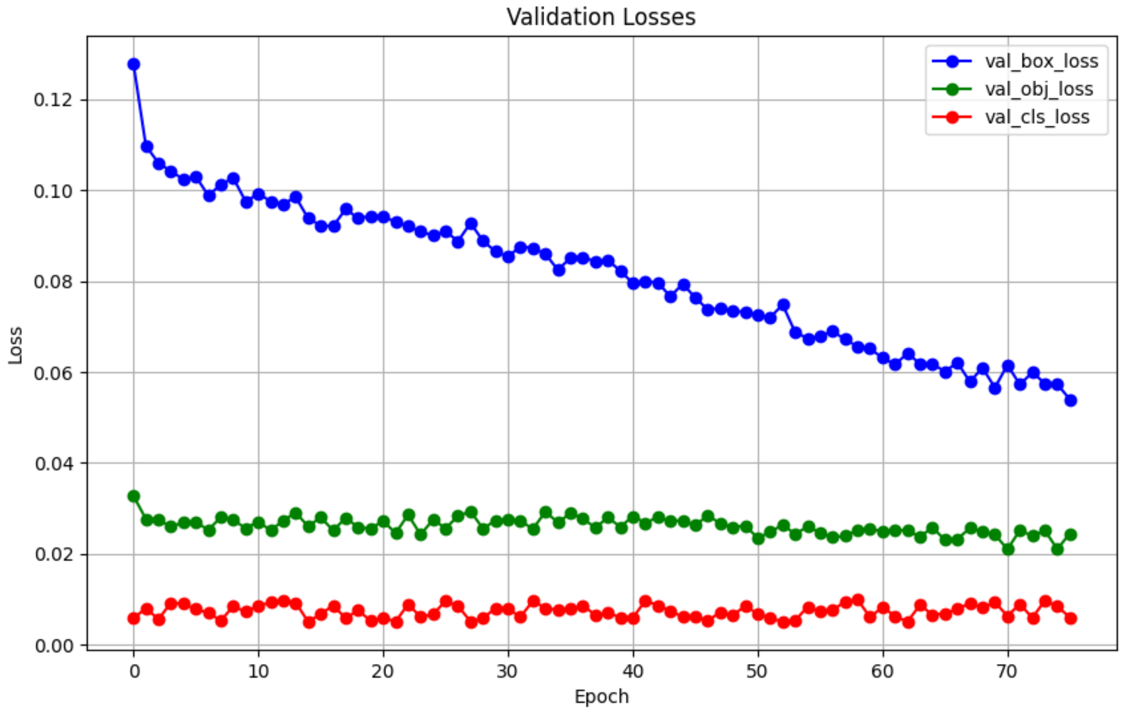
Resizing

RCNN models too expect images of a certain size as input. It is required to resize the images to match the input dimensions required by the model. We resized images to 640 x 640. Resizing helps ensure that the model processes the entire image and captures objects at different scales. Many object detection models require input images to have a fixed size. Resizing ensures that all images fed into the model have the same dimensions, which is necessary for the model to process them efficiently and make predictions accurately. Resizing images to a smaller size reduces the memory and computational resources required to process them. It helps to train faster

1.2.3 Training and validation loss curves

Losses were greater compared to YOLO Based Model.

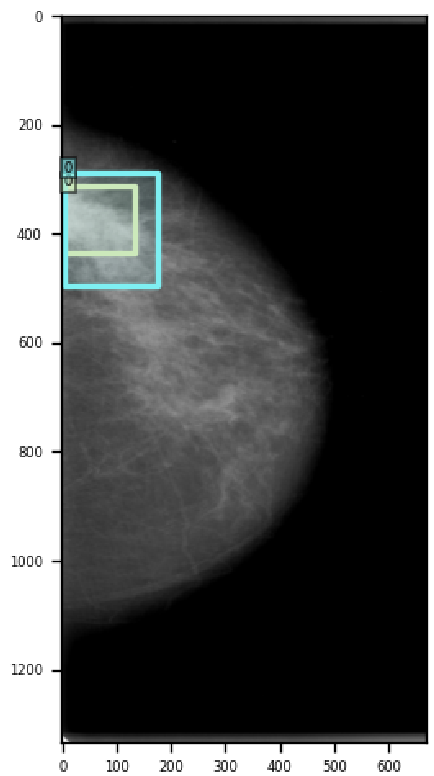


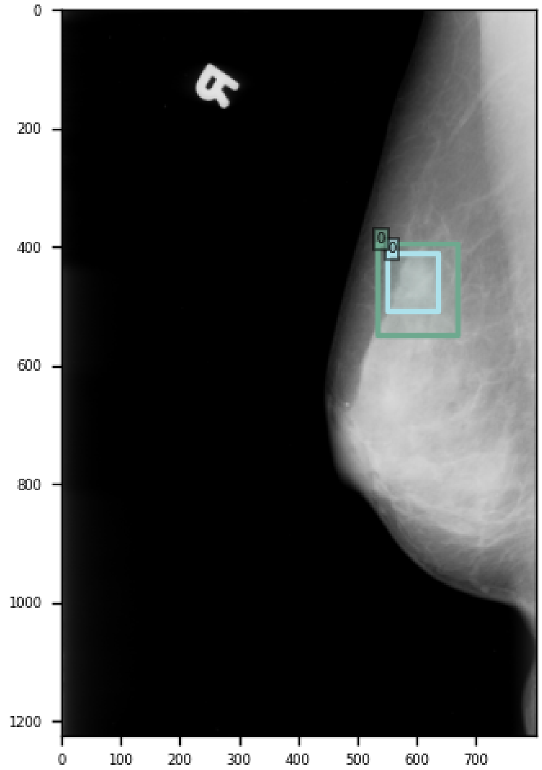


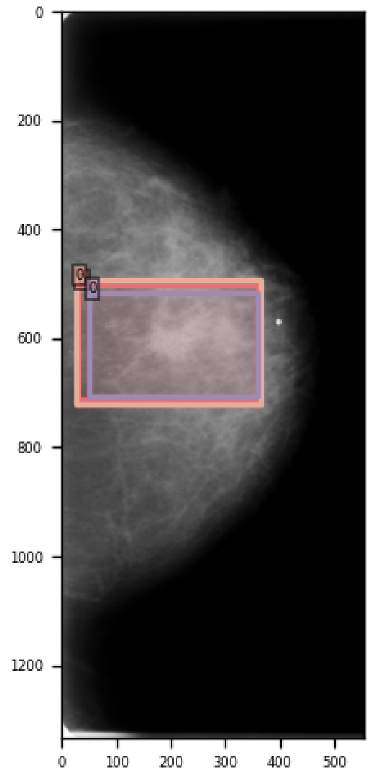
2 Transformer Based Models

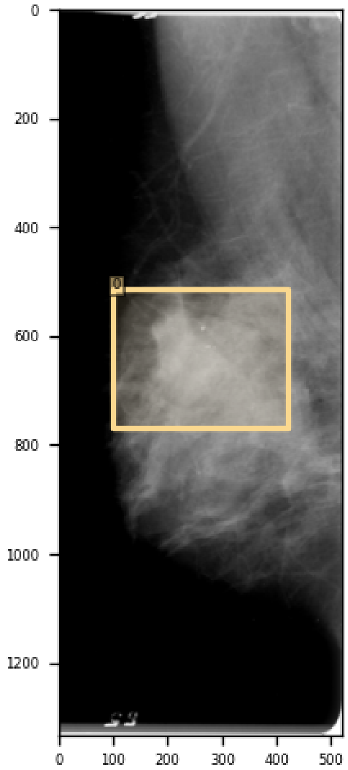
2.1 DINO

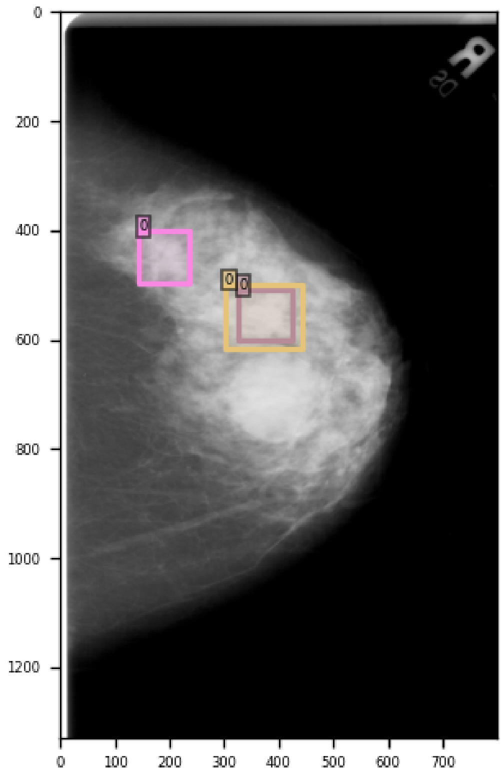
2.1.1 Data Visualisation




0
200
400
600
800
1000
1200
0
100
200
300
400
500
600
700


0
200
400
600
800
1000
1200
0
100
200
300
400
500


0
200
400
600
800
1000
1200
0 100 200 300 400 500
0


0
200
400
600
800
1000
1200
0 100 200 300 400 500 600 700
0
0
0

2.1.2 Image Preprocessing

Resizing

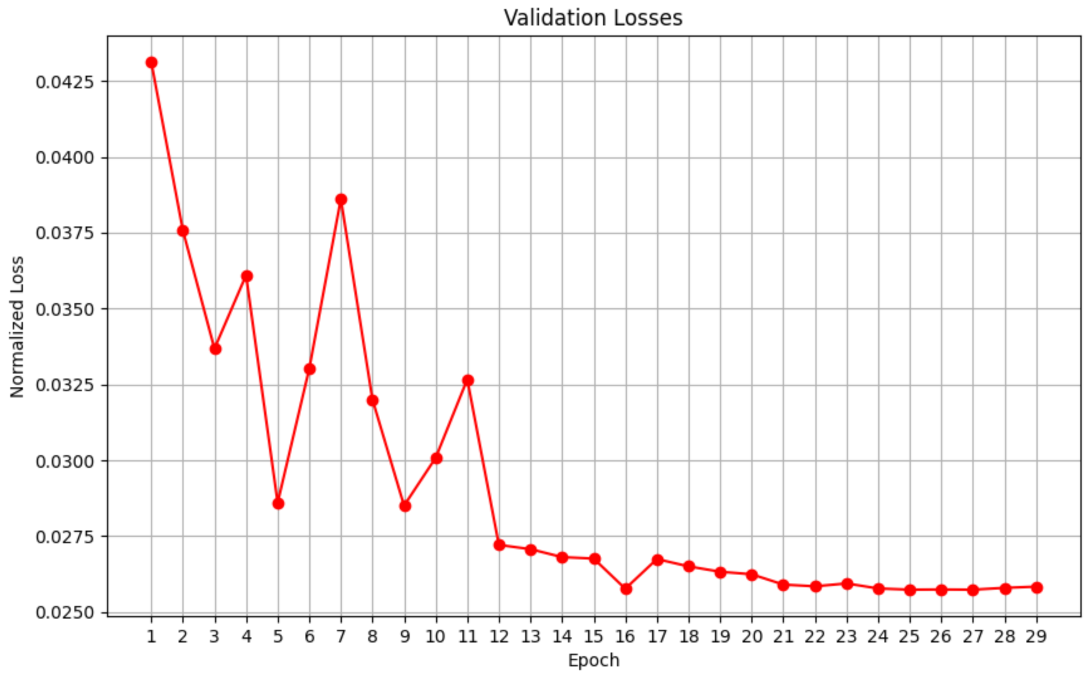
DINO models often work well with square images. So we resized our images to an appropriate square of 640 x 640.

Normalization

We normalize the pixel values of the images. This involves scaling the pixel values to be in the range [0, 1] or [-1, 1]. Normalization helps the model converge faster during training.

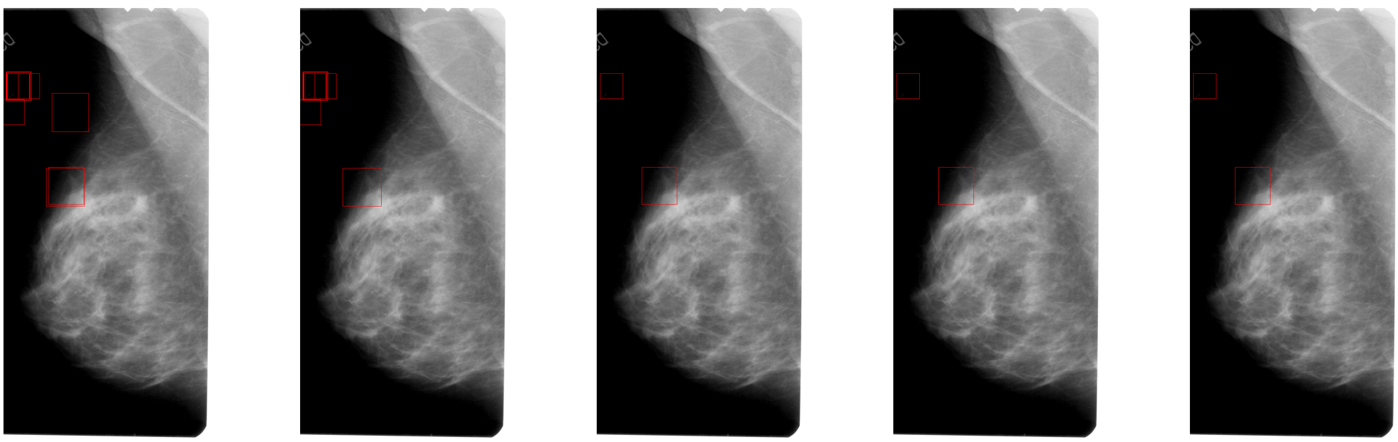
2.1.3 Training and validation loss curves

The losses are significantly smaller when compared to convolution based models. This might be due to the fact that they compute losses in different ways. Anyhow, other metrics like mAP50 were significantly better for the DINO model.



2.1.4 Non-Maximum Suppression

For the same image DINO detected 13 malignant boxes, though all very close to each other as depicted for threshold 1



(a) Threshold : 1 (13 malignants)

(b) Threshold : 0.5 (8 malignants)

(c) Threshold : 0.25 (2 malignants)

(d) Threshold : 0.1 (2 malignants)

(e) Threshold : 0 (2 malignants)

Figure 2: NMS

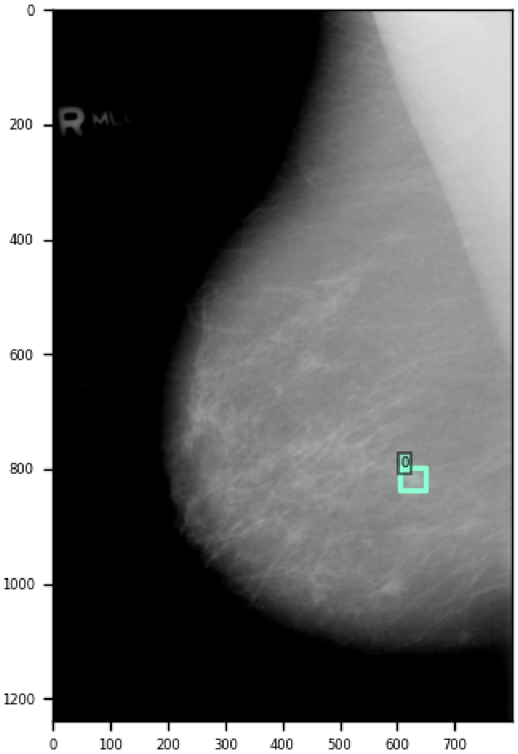
2.1.5 Grad-CAM/Attention Maps visualizations

3 Comparison between Models

DINO is the best performing model out of the 3 we tried out.

DINO performed better on the basis of FROC score, mAP50, map50:95. Visualising the predicted bounding boxes also showed that DINO was performing better than yolov5.

For instance tumour in the following mammogram was detected by DINO, however both YOLO and RCNN could not detect it.



4 Link to models

https://drive.google.com/drive/u/0/folders/1GmqoUuLTW-sFuusVF4FgKt$_BF0PMq8E6$