BREAST CANCER DETECTION IN MAMMOGRAMS USING DEEP NEURAL NETWORKS

Anvit Mangal 2016135
Surabhi S Nath 2016271

BTP report submitted in partial fulfillment of the requirements
for the Degree of B.Tech. in Computer Science & Engineering
on 16th November, 2019

BTP Track: Research Track

BTP Advisor
Prof. Chetan Arora, IIT Delhi
Prof. Subhashis Banerjee, IIT Delhi

Indraprastha Institute of Information Technology
New Delhi

Student's Declaration

I hereby declare that the work presented in the report entitled **Breast Cancer Detection in Mammograms using Deep Neural Networks** submitted by me for the partial fulfillment of the requirements for the degree of *Bachelor of Technology* in *Computer Science & Engineering* at Indraprastha Institute of Information Technology, Delhi, is an authentic record of my work carried out under guidance of **Prof. Chetan Arora**. Due acknowledgements have been given in the report to all material used. This work has not been submitted anywhere else for the reward of any other degree.

Anvit Mangal **Place & Date: IIIT Delhi, 16th November 2019**
Surabhi S Nath

Certificate

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Prof. Chetan Arora **Place and Date: IIT Delhi, 16th November 2019**

Abstract

Breast cancer has become the most common form of cancer in Indian, recently having overtaken cervical cancer in urban cities. Immense research has been carried out on breast cancer and several automated machines for detection have been formed, however, they are far from perfection and medical assessments need more reliable services. Also, very little research has been performed on Indian datasets, which are significantly different than the available foreign resources. Indian breasts are more dense and hence differ in texture, lesion size and composition. Our works aims to reproduce the state of the art results reported by researchers using deep learning approaches to automate breast cancer detection and extend them to build deep learning breast cancer detection networks which are more specific to Indian breast types and alongside use metadata of age, breast density, past history and other available information to enable more accurate judgement and treatment. Working in close collaboration with a radiologist from AIIMS Delhi, our work develops alternative object detection frameworks, ensemble and transformer networks to improve and enhance both classification and detection of breast lesions. Various analysis technique have been applied to measure performance and conduct thorough investigation of our experiments. Parts of our code can be found here: `https://github.com/anvitmangal/breastcancerdetection/`

Keywords: Breast Cancer, Mammograms, Deep learning, Neural Networks, Biomedical Applications, Object Detection, Faster-RCNN, YOLO V3, RetinaNet, Expert ensemble network, Photometric transformer

Acknowledgments

We thank Prof. Chetan Arora for giving us this project and providing us the guidance and support for all our work. We also thank Prof. Subhashis Banerjee for the insightful discussions and his continuous advice. We express our gratitude towards Dr. Krithika Rangarajan for constantly being there to help us and enabling us to learn and appreciate the field.

Work Distribution

All work was evenly distributed among us. Most work was done together, with certain sub-tasks being individually assigned to enable us to work parallely and hence more efficiently. Other than us, Dr. Krithika Rangarajan (AIIMS Delhi, IIT Delhi) was responsible for annotating the Indian dataset of breast cancer mammograms obtained from AIIMS Delhi.

Contents

Chapter 1

Introduction

1.1 Problem Statement

Breast cancer impacts nearly 1.5 million women in all over the world each year and causes the greatest number of cancer-related death of women. Trends say 1 in every 8 women will be diagnosed with breast cancer in her lifetime. Breast cancer has become the most common form of cancer in Indian urban cities, recently having overtaken cervical cancer and 2nd most common in rural India. There has been a shocking increase in the number of cases affecting women at a much younger age than 25 years ago. Most women today in India fall pray to breast cancer at around their 40s, which is much younger than when cancer hits in foreign countries.

The major causes of breast cancer are mostly genetic - damaged DNA and family history. However, other risk factors can be lifestyle or environmental related, like alcohol consumption - studies reveal that women who consume 3 drinks a day have a 1.5 times higher chance of being affected, obesity, hormonal treatments - an increased level of estrogen due to hormone replacement therapy of birth control pills can link to breast cancer, sedentary life without physical exercise. Other reasons such as bearing a child late in life or improper breastfeeding may also be responsible.

However, the biggest reasons are the lack of awareness, treatment and screening methods. Non availability of expert radiologists and diagnostic centres and delay in extending necessary care is a major concern. In an attempt to assist this growing cause, we aim to develop deep learning models to detect suspicious lesions and hence provide timely and effective diagnosis.

One may argue that immense research has been carried out on breast cancer and similar learning machines have been developed in the past. However, we found that although these machines exist, they are far from perfection and medical assessments need more reliable services. Also, very little research has been performed on Indian datasets. Most results and infact all datasets too, are only available for foreign patients. We notice several variations in the breast development and structure of Indian women including age difference when cancer initiates and breast

tissues - Indian breasts are more fibrous, hence whiter and denser. We therefore feel that the same networks when applied on Indian cases cannot yield accurate information.

Our aim is to build deep learning breast cancer detection models which are more specific to Indian breast types and alongside use metadata of age, breast density, past history and other available information to enable more accurate judgement and treatment.

1.2 Terminology

Some essential breast cancer terminology are as follows:

1. **Mammography** - Mammograms are screening techniques like the x-ray. Mammogram images are used for initial detection of lumps and lesions in the breast. Tissues appear as white and gray regions represents the fat. Mammograms are captured in 2 views - MLO (mediolateral-oblique) and CC (Cranial-Caudal), for both the breasts.

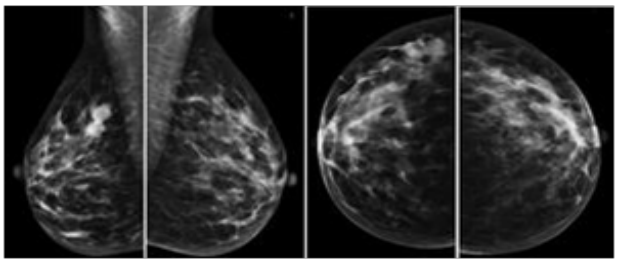


Figure 1.1: MLO and CC views of Left and Right Breast

2. **MRI scan, ultrasound** - These are other scanning methods used for breast cancer evaluation. An ultrasound is generally done post the mammogram if a suspicious lesion is found. It helps distinguish and differentiate between benign fluid-filled cyst from a solid mass.

3. **Biopsy** - A sample of tissue is surgically removed and the cells are analyzed. This can show whether the cells are cancerous, and, if so, which type of cancer it is. This is the ultimate proof of breast cancer.

4. **Benign and Malignant** - Lesions can be either benign, malignant or normal. When the breast image does not indicate the presence of any lesion, it is considered a normal breast. Benign lesions are non cancerous lesions while malignant lesions are cancerous.

5. **Mass and calcification** - Masses and calcifications are two most commonly occuring abnormalities in the breast. Calcifications are calcium deposits which occur as small dotted specs in the breast. Masses are lumps of tissue that are dense at the centre and lesser dense at the peripheral. Both mass and calcifications can have benign and malignant

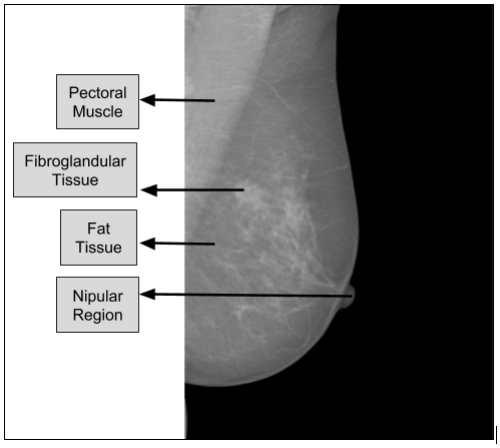


Figure 1.2: Sample Breast Mammogram

variants. Benign calcification types include ourse, popcorn calcification, scattered, lucent centred, milk of calcium; and malignant variants are fine and amorphous calcification. Spiculated masses distort the breast architecture and are very likely to be cancerous.

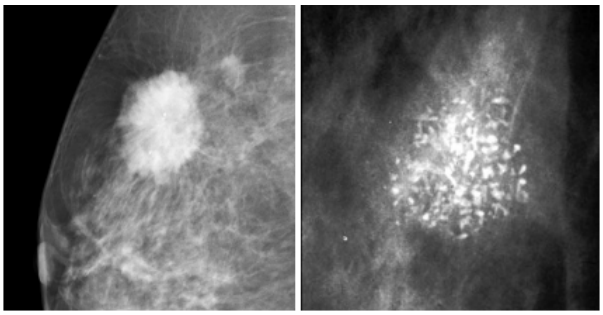


Figure 1.3: Left: Mass, Right: Calcification

6. **BIRADS scoring** - The BIRADS scoring of 0 to 6 is a scoring scale which indicates the level of cancer.
    - BIRADS 0 - Incomplete: Need more information, ultrasound is suggested. Ultrasound clears most of the BIRADS 0 cases.
    - BIRADS 1 - Normal breast, negative
    - BIRADS 2 - Benign
    - BIRADS 3 - 95% Benign
    - BIRADS 4 - Suspicious (5 to 95% Malignant)
        - 4A - Low suspicion of Malignancy
        - 4B - Moderate suspicion of Malignancy
        - 4C - High suspicion of Malignancy
    - BIRADS 5 - 95% Malignant

- BIRADS 6 - Biopsy verified Malignant

This BIRADS scoring can further be classified into Actionable and Non-actionable classes, with BIRADS 0, 3, 4 and 5 considered actionable and 1, 2 considered non-actionable.

Chapter 2

Literature Review

Research on the topic revealed that even though the breast cancer detection problem has been an important problem studied for multiple decades, the older approaches have majorly used hand-crafted features like tissue densities, age and other properties of the breast and have applied machine learning classifiers on the features. Deep learning automates the feature extraction step and ensures it extracts the most relevant features as desired by the task. CNN based approaches have been implemented for breast cancer but it only performs classification and not detection. Classification networks are commonly fed image patches and a binary classifier decides whether the fed patch is benign, or malignant. However, for assisting and aiding the field of cancer diagnosis, we need to be able to give a whole breast image and accurately mark out the regions containing suspicious lesions. This is the detection or localization task. Deep learning based solutions to lesion localization are still being explored with only few mainstream publications setting the benchmark standards.

Our task is to successfully both detect and localize lesions in mammograms and further also classify them as malignant or benign.

A recent paper published in the Nature Scientific Reports: Detecting and classifying lesions in mammograms with Deep Learning, by Dezso Ribli et. al. was published in 2018 [1]. This paper proposed a CAD system based on one of the most successful object detection frameworks, Faster R-CNN. The system detects and classifies malignant or benign lesions on a mammogram without any human intervention. The proposed method set the state of the art classification performance on the famous public INbreast digital breast mammogram database, AUC = 0.95. The approach used by the authors in the paper achieved 2nd place in the Digital Mammography DREAM Challenge with AUC = 0.85 on the DREAM dataset. When used as a detector, the system reached high sensitivity with very few false positive marks per image on the INbreast dataset. The authors have used ROC curves to find AUC to evaluate the classification performance and an FROC curve which plots sensitivity against the number of average false positive marks per image to validate the detections. We have considered this paper as the baseline for

all our work and have worked on first reproducing this paper, followed by proposing our own ensemble network for improved performance in breast cancer detection.

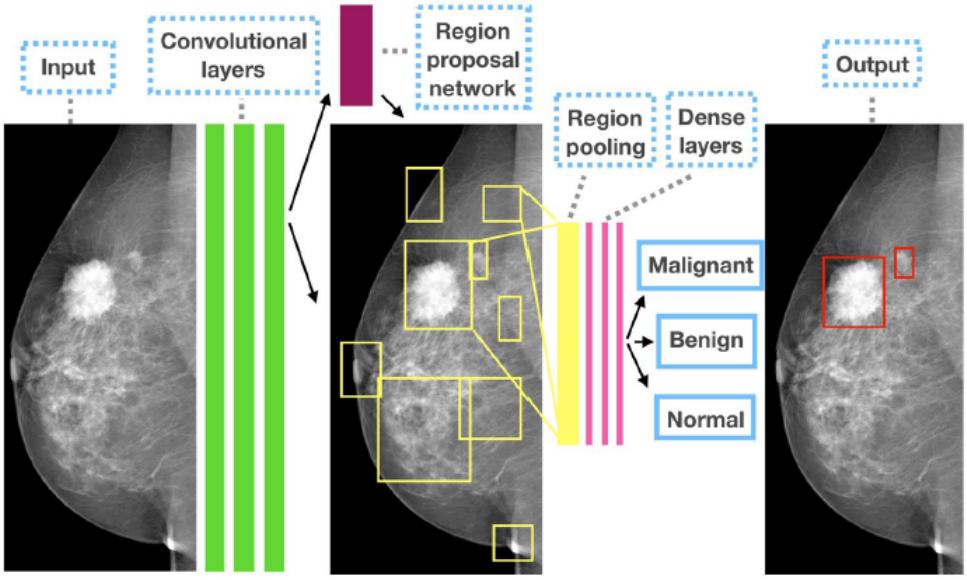


Figure 2.1: Outline of FasterRCNN based CAD model

We explored all the popular object detection frameworks, including Faster-RCNN, YOLO and RetinaNet in order to understand their functioning, architectures, advantages, disadvantages, tradeoffs and performance on common benchmark datasets like Pascal VOC, COCO, etc. We also coded object detection networks from scratch to learn network definitions thoroughly.

As a sanity check for the scores we got for the Retinanet model trained using the DDSM, INBreast and AIIMS dataset, we explored multiple techniques that generate saliency maps, or regions where the object detection model is focusing while predicting the objects in the image. For this reason, we searched for many papers that propose methods to generate saliency maps. Learning Deep Features for Discriminative Localization, Zhou et. al. proposes a method Class Activation Map or (CAM) which uses Global Average Pooling to generate saliency maps. A similar and a more robust method is suggested in the paper Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization. We also explored the perturbation techniques as suggested by the paper Visualizing and Understanding Convolutional Networks by Fergus.

To solve the problem of poor results on Dense breasts, we explored techniques which normalised images by learning parameters for the intensity transformation on the fly using a Neural Network. One such method was proposed in the paper Photometric Transformer Networks and Label Adjustment for Breast Density Prediction. The paper presented the following problem: the grading of the breast density is sensitive to:

$$h(x(i,j),S) = \begin{cases} u + s_0(x(i,j) - u) & \text{if } x(i,j) \in (-\infty, u) \\ u + \sum_{l=1}^{k-1} s_l t + s_k(x(i,j) - \min(T_k)) & \text{if } x(i,j) \in T_k \\ u + \sum_{l=1}^{k} s_l t + s_{K+1}(x(i,j) - v) & \text{if } x(i,j) \in [v, \infty) \end{cases}$$

Figure 2.2: H Function

1) Distribution of pixel intensity 2) uncertain grading policies of mammogram readers To solve both of the above issues, they propose: 1) Photometric Transform Network: to adaptively normalise the mammograms, learns parameters for the transform on the fly and can be fit anywhere in a network. 2) Label Distillation: intended to mitigate the grading variations and use a pseudo-label technique Some background to the density of breasts and its effects: Breasts classified into 2 types: 1) Dense 2) Fatty Readers should be careful when analysing the dense breasts as there are more dense parenchymal patterns and thus suspicious malignant lesions can be hidden, leading to more false-negatives.

Chapter 3

Materials and Methods

3.1 Datasets

Several famous Breast mammogram datasets are publically and freely available which we acquired and used for our research work.

DDSM Dataset - The DDSM (Digital Database for Screening Mammography) dataset is one of the most famous databases for breast mammographic research. It is a resource popularly used by the entire mammographic image analysis research community. Primary support for this project was a grant from the Breast Cancer Research Program of the U.S. Army Medical Research and Materiel Command. The Massachusetts General Hospital, the University of South Florida, and Sandia National Laboratories have also contributed. Additional cases were provided from Washington University School of Medicine. The dataset contains nearly 2500 studies with 12 volumes of normal images, containing 695 cases; 15 volumes of cancerous, containing 855 cases; 14 volumes benign, containing 870 cases; and 2 volumes of benign without callback, containing 141 cases.

Each case has between 6 to 10 files:

- 4 images of Right MLO, CC views and Left MLO, CC views in compressed lossless .LJPEG format.
- Upto 4 .OVERLAY files - OVERLAY file exists for an image only if there exists a lesion (benign or malignant). Hence normal images will not have any .OVERLAY files. This file contains count of number of abnormalities in the breast, Assessment (BIRADS score), Subtlety, Pathology (Benign or Malignant) and the Boundary of lesion in the form of chain code.
- One .ics file - mentioning all metadata: date of study, patient age, film type, density, pixels, resolution of each image and whether each image is accompanied with overlay or not.

- One 16 bit .pgm file

Decompression from the lossless .LJPEG format was a big challenge. After trying from various sources, we were finally able to convert to .JPEG, although it was not the best option since .JPEG format is a lossy format and conversion from and to .JPEG causes loss in image quality, but there was no other format to convert .LPEG into directly. We then realized that after format conversion, many (nearly 1/8th) images were completely black with no visibility and colour. These images had to be disregarded before training on the images.

Another major challenge was decoding the chain code to get bounding box coordinates. Since we used PASCAL VOC format, we needed to feed .xml files as input. For this, we had to create the .xml files ourselves. Thus, the bounding box was extracted from the chain code and following that xml files had to made in the required format with the desired structure - name, type, objects, bndboxs with xmin, xmax, ymin, ymax coordinates.

Also, the mammograms were really big, nearly 4000 x 4000 pixels hence they had to be resized to a suitable size before training on them. Plus, the images did not have enough contrast hence all images were mode adjusted using pixel clipping.

INbreast Dataset - The INbreast database is a mammographic database, with images acquired at a Breast Centre, located in a University Hospital (Hospital de So Joo, Breast Centre, Porto, Portugal). INbreast has a total of 115 cases (410 images), out of which 90 cases are of women from both breasts (4 images per case) and 25 cases are from mastectomy patients (2 images per case). Several types of lesions (masses, calcifications, asymmetries, and distortions) are included. Accurate contours made by specialists are also provided in XML format. The dataset was acquired on request from the owners.

The files included were the following:

- Dicom images - 410 dicom images
- ROIs - .roi for 343 images containing lesions (non normal images)
- XMLs - .xml files for 343 images containing lesions (non normal images)
- A .csv file documenting all information for all images - Patient ID, Patient age, Laterality, View, Acquisition date, Filename, ACR and Birads score.

All images were mode adjusted to improve contrast by clipping all pixel intensities to 300 below the mode and 500 above the mode. All pixels were then rescaled into 0-255, since our images had 16 bit depth. Further, the images were resized to appropriate size of 2100 x 1700.

Indian AIIMS Dataset - A professional radiologist from AIIMS, is part of our project. She provided us with the AIIMS dataset. However, since these images were unannotated, she annotated nearly 3000 images for us. Annotations were done in three ways - as per the BIRADS

scoring, as per mass or calcification, as per actionable or non-actionable. The images were in .dicom and were converted to .png. Annotations were obtained in an xml file in PASCAL VOC format. We experimented with all sizes of lesions, small masses, large masses, calcification spots, clusters and node lesions.

3.2 Background Work

Our task was to reproduce the figures pronounced by the authors in [1] for classification and detection on various networks including FRCNN,

3.2.1 Faster RCNN Framework

The localization problem for the lesions demands an algorithm which would not only label the class of the lesion, i.e. label it as benign or malignant but also draw a bounding box around the lesion. Several object detection and localisation algorithms and techniques have been introduced in the last few years. Out of these, the Faster-RCNN method proposed by Ross Girshick et. al. is the most commonly and widely used object localization algorithm.

Object Detection Task

The difference between object detection algorithms and classification algorithms is that in detection algorithms, we try to draw a bounding box around the object of interest to locate it within the image. Also, we might not necessarily draw just one bounding box in an object detection case, there could be many bounding boxes representing different objects of interest within the image and we would not know how many beforehand.

The major reason why we cannot proceed with this problem by building a standard convolutional network followed by a fully connected layer is that, the length of the output layer is variable, this is because the number of occurrences of the objects of interest is not fixed. A naive approach to solve this problem would be to take different regions of interest from the image, and use a CNN to classify the presence of the object within that region. The problem with this approach is that the objects of interest might have different spatial locations within the image and different aspect ratios. Hence, we would have to select a huge number of regions and this could computationally blow up. Therefore, algorithms like YOLO, RCNN, Fast-RCNN, Faster-RCNN etc have been developed to find these occurrences and find them fast.

Faster RCNN has 2 networks: the region proposal network (RPN) and another network to give the final classification and bounding boxes for the objects. The RPN proposes several possible regions (anchors) where the object may reside in and also ranks these anchors on the basis of the probability of the object residing in the region. These regions are then fed to the next network to choose the most favourable region. Faster RCNN is better than the prior object

detection counterparts like Fast RCNN which uses Selective Search to propose regions. RPN is much faster than the selective search algorithm and produces much better results. In the default configuration, there are 9 anchor boxes on each position, with 3 different sizes, each of them having 3 different scales.

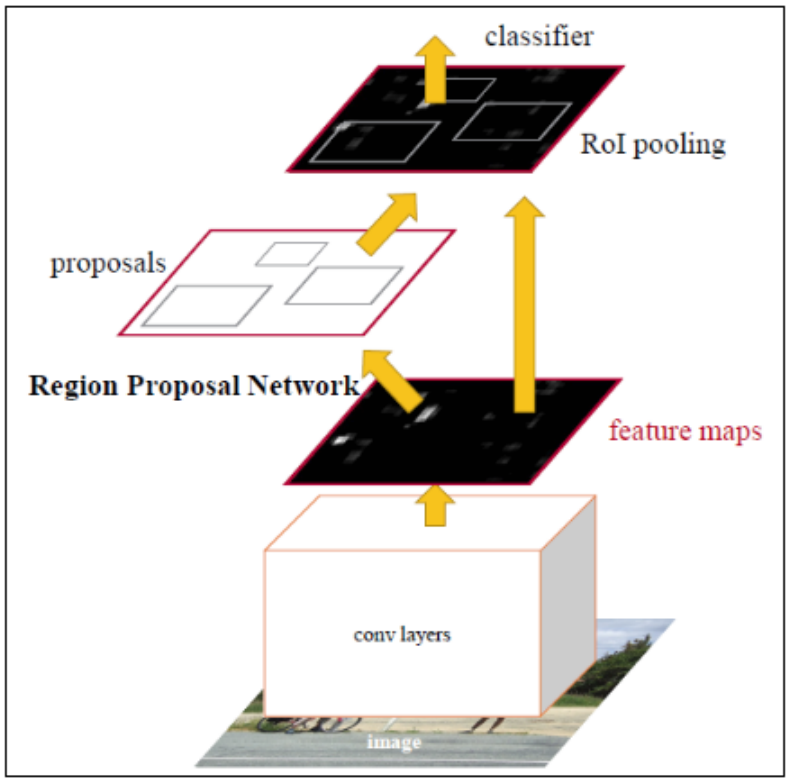


Figure 3.1: Faster RCNN Architecture

The Region Proposal Network predicts the probability of an anchor being background or foreground, and to further refine the anchor. The output of the RPN are a bunch of bounding boxes or regions as well as an assigned class label (foreground/background) to each one of them. These labels and bounding boxes are then fed to the next network which finally examines the correctness of these labels and bounding boxes using classifier and regressor respectively. The anchors having the higher overlaps with ground-truth boxes are labelled as foreground, the ones with lower overlaps as background.

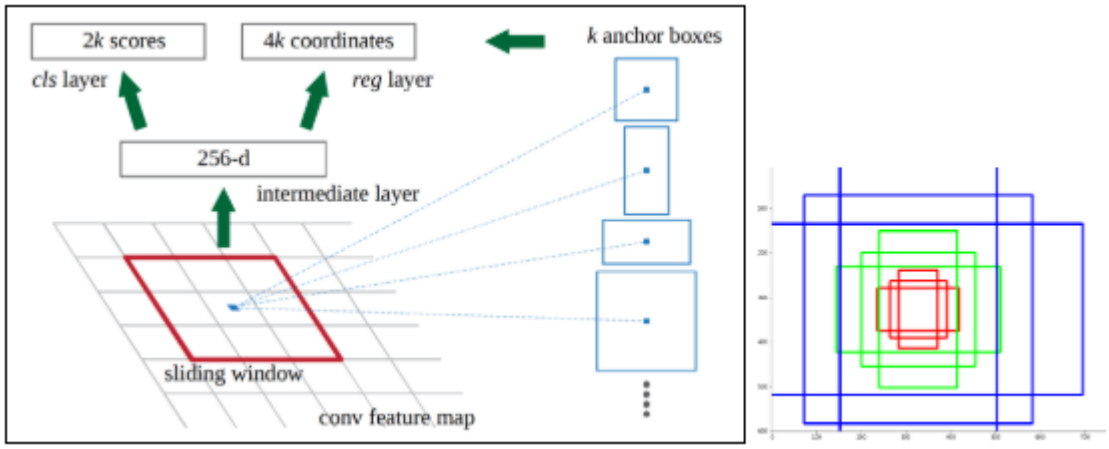


Figure 3.2: RPN Network (left) and Anchor Boxes (right)

The background class has to be ignored in this case as there is no ground truth bounding boxes for the background class. The paper uses smooth-L1 loss on the position (x, y) of top-left the box, and the logarithm of the heights and widths.

One important characteristic of the regions proposed by the RPN is that they are of different sizes and different sized regions means different sized CNN feature maps. To tackle this, ROI pooling layer was introduced which divides the feature maps into a fixed number (say k) of roughly equally sized regions. Max-pooling is then applied on these regions to give CNN feature maps of the same sizes.

3.2.2 YOLO V3 Framework

YOLO or You Only Look Once is one of the most successful object detection algorithms in recent times. We used YOLO V3 which is a modified version of its previous counterparts: YOLO and YOLO V2.

YOLO lacked the ability to capture fine grain details hence YOLO V2 incorporated identity mapping, and concatenated feature maps from from a previous layer to capture low level features. Still, YOLO V2 lacked some of the elements like residual blocks, skip connections and upsampling which are part of other object detection algorithms. YOLO V3 remedied this issue by incorporating all these features in its architecture.

	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
1×	Convolutional	32	1 × 1	
	Convolutional	64	3 × 3	
	Residual			128 × 128
	Convolutional	128	3 × 3 / 2	64 × 64
2×	Convolutional	64	1 × 1	
	Convolutional	128	3 × 3	
	Residual			64 × 64
	Convolutional	256	3 × 3 / 2	32 × 32
8×	Convolutional	128	1 × 1	
	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
8×	Convolutional	256	1 × 1	
	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
4×	Convolutional	512	1 × 1	
	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

Figure 3.3: YOLO V3 Initial 53 Layers

YOLO V3 uses a variant of Darknet which originally has 53 layers. The 53 layers for the feature extraction task are then stacked with an additional 53 layers for the detection task to get a 106 layered convolutional neural network.

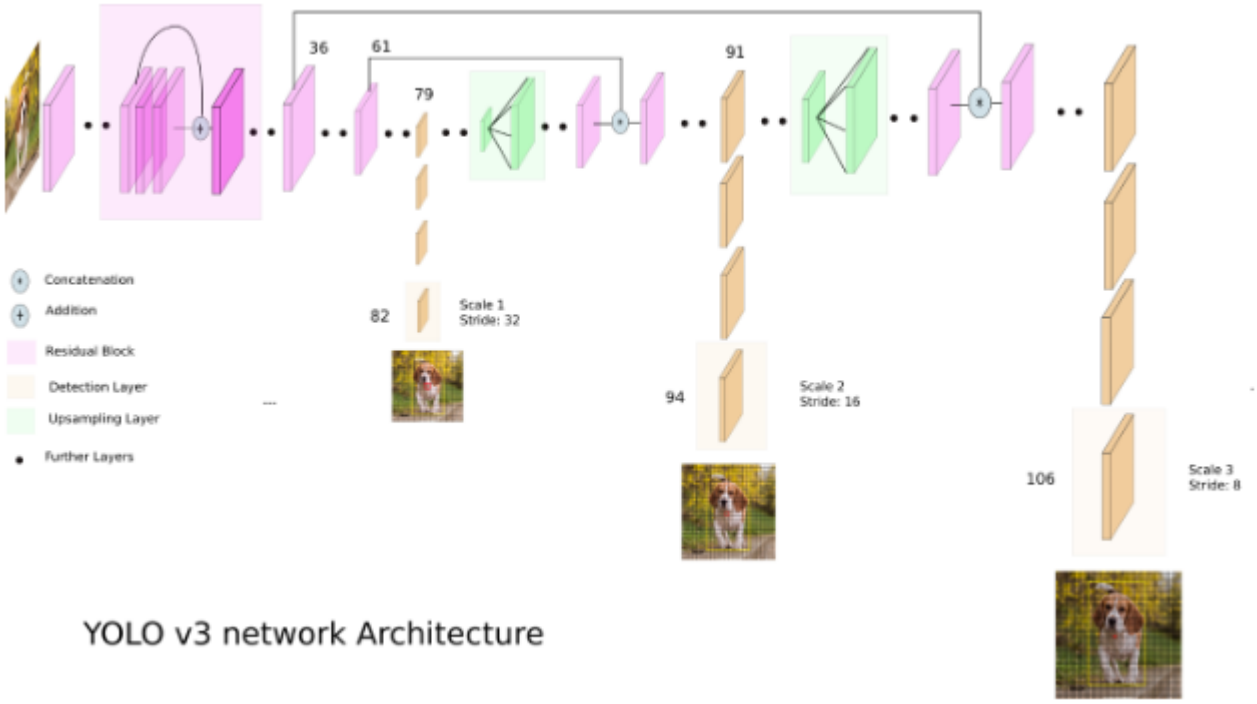


Figure 3.4: YOLO V3 Complete Architecture

Multi-scale Detection

YOLO V3 produces predictions at 3 different scales. The output is generated by applying a 1x1 kernel to the feature maps. The 416 x 416 input image is downsampled by applying convolutional layers till the 81st layer.

- The first predictions are made by the 82nd layer, giving an output feature map of size 13 x 13 x 255 which is obtained by applying the 1 x 1 kernel to a 13 x 13 feature map.
- The second detection is made by the 94th layer, giving output feature maps of size 26 x 26 x 255. This is obtained by applying a few downsampling convolutions to the feature map at layer 79 and then upsampling it 2x to obtain a feature map of size 26 x 26. This is then depth concatenated by the feature map at layer 61.
- The third and final detection is made by the 106th layer to obtain output feature maps of size 52 x 52 x 255. Similar to what happens at the second scale, this feature map is obtained by applying a few downsampling convolutions to the feature map at layer 91 and then depth concatenating with the feature map at layer 36.

Anchor Boxes

YOLO V3 applies 9 anchor boxes, 3 for each scale. Anchor boxes are used to to fit the approximate sizes of objects in the dataset. These anchor boxes are calculated using the K-Means clustering algorithm, where all the objects in the dataset are fit into 9 clusters, to get 9 sizes: height and width, for the bounding boxes. The largest 3 anchors are used to predict the boxes at the first scale, the next largest 3 anchor boxes for the second scale and the smallest 3 boxes are used for the third scale.

The network predicts 4 coordinates for each bounding box, tx, ty, tw, th. If the cell is offset from the top left corner of the image by (cx, cy) and the bounding box prior has width and height pw, ph, then the predictions correspond to bx, by, bw and bh. The width and height of the box are predicted as offsets from cluster centroids. The center coordinates of the box are predicted relative to the location of filter application using a sigmoid function. YOLOv3 predicts the 10x number of bounding boxes (10,647) as compared to 845 for YOLOv2. This is the reason why YOLOv3 is slower than YOLOv2 and YOLO.

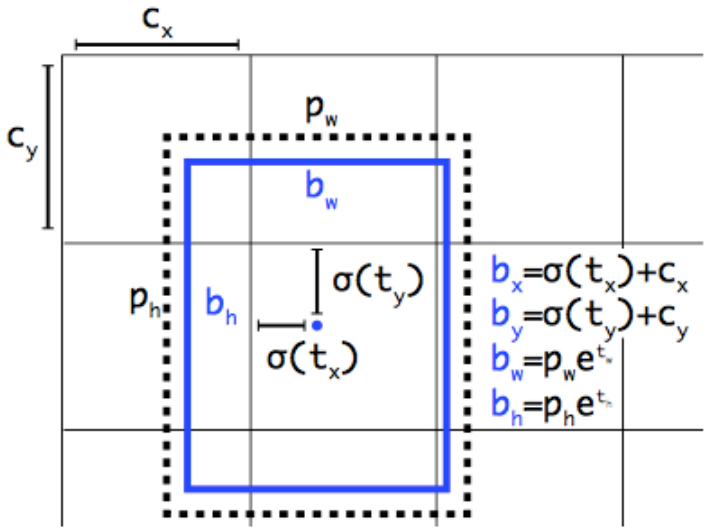


Figure 3.5: YOLO Predictions

Loss Function

The YOLO V2 loss function consists of 5 terms. The first 2 terms are identical in YOLO V3, while for the last 3 terms, cross entropy loss is used instead of squared error.

$$\lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left(x_i - \hat{x}_i\right)^2 + \left(y_i - \hat{y}_i\right)^2$$

$$+ \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left(\sqrt{w_i} - \sqrt{\hat{w}_i}\right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i}\right)^2$$

$$+ \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{obj}} \left(C_i - \hat{C}_i\right)^2$$

$$+ \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \mathbb{1}_{ij}^{\text{noobj}} \left(C_i - \hat{C}_i\right)^2$$

$$+ \sum_{i=0}^{S^2} \mathbb{1}_{i}^{\text{obj}} \sum_{c \in \text{classes}} \left(p_i(c) - \hat{p}_i(c)\right)^2$$

Figure 3.6: YOLO Predictions

- The first term penalizes the coordinates of the centre of the predicted bounding box.
- The second term penalizes the height and width of the predicted bounding box.
- The third term penalizes the objectness of the predicted bounding boxes which are responsible for predicting objects (scores close to 1)
- The fourth term penalizes the objectness of the predicted bounding boxes which are responsible for predicting no objects (scores close to 0)

- The fifth term penalizes the class prediction of the predicted bounding boxes.

3.2.3 RetinaNet Framework

RetinaNet is an object detection framework which is commonly used today for several applications. RetinaNet has been able to perform well on several benchmark datasets. The two major contributions of the RetinaNet are the Feature Pyramid Network and Focal Loss.

Feature Pyramid Network

The Feature Pyramid network is used to detect objects at different scales and for small object detection. This Neural Network is made keeping in mind the performance rather than speed. FPN is based on the fact that deeper the convolution, more is the semantic value of the features it detects, due the increased depth, the resolution of the part of the image it captures is less. The Lower layers are semantically very strong but it cannot precisely detect the object, so skip layers similar to ResNet are added. This is also inspired the use of ResNet as its base network for the initial feature extraction i.e. the bottom-up pathway. As we move up, the spatial dimension is reduced by 1/2 (i.e. double the stride). In the top down pathway, upsampling is done using nearest neighbour upsampling and 1x1 convolutional map is used and then 3x3 convolutional map is applied over it. FPN is a feature detector and over which an object detector is added. FPN in Retinanet provides a top-down pathway to construct higher resolution layers from a semantic rich layer.

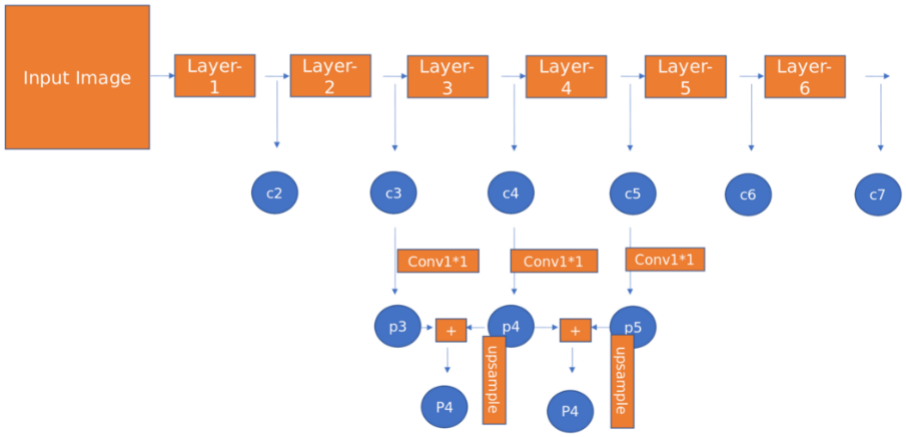


Figure 3.7: RetinaNet Feature Pyramid Network

Retinaet divides the network into 2 subnets - one for classification and one for detection. The network generates A = 9 translation invarient anchor boxes, through 3 pyramidal levels with aspect ratios 1:1, 1:2, 2:1. It uses a class-agnostic bounding box regressor, with fewer parameters yet achieves equivalent performance. Also, the two subnets share the same structure but use different parameters.

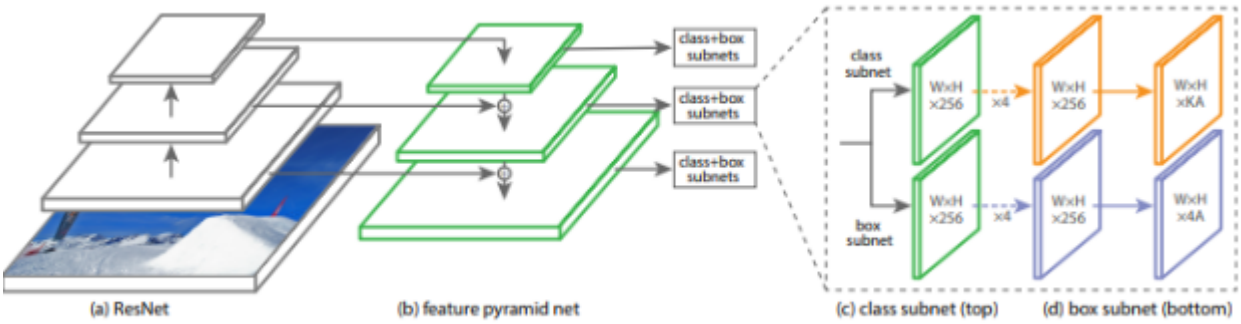


Figure 3.8: RetinaNet 2 Subnet Architecture

The focal loss employed in RetinaNet is similar to cross entropy loss but it scales the loss so that the easier examples contribute less to the loss and the difficult examples contribute more. It introduces one more hyperparameter - gamma, which when 0 implied the normal CE loss equation. As a result of this, the model is tuned to identify the difficult objects more accurately.

$$C(p, y) = -\sum_i y_i (1 - p_i)^\gamma \ln p_i$$

Figure 3.9: RetinaNet Focal Loss

3.2.4 Why YOLO V3 and Retinanet

Our task of Breast cancer detection involves detecting both large masses and small calcifications. Faster-RCNN has good results on detecting large objects, while for smaller objects Faster-RCNN fails to perform well.

YOLO and YOLO V2 were poor in detecting smaller objects too, but in YOLO V3 displays a reverse trend. YOLO V3 with its multi-scale prediction method has better AP scores for small objects than Faster-RCNN, YOLO V2 and YOLO. Also, YOLO V3 is a single stage object detection algorithm, and is extremely fast; 22ms inference time for 320 x 320 input, as compared to 172 ms for Faster-RCNN. This is the the reason why utilized YOLO V3 for our go to model for breast cancer detection.

Similarly, we experimented with Retinanet because mammogram datasets present features which are to be picked at multiple scales. Masses could be large and small while calcifications can be large when clustered together and extremely small for present in isolation as spots. Hence, RetinaNet can prove to be useful for such an application.

3.2.5 Training

Both YOLO V3 and RetinaNet were trained on mammogram datasets using several experiments and analysis techniques.

YOLO V3

Training using DDSM and AIIMS

We trained YOLO V3 using a combined training dataset of DDSM and AIIMS which had 2700 mammograms. Heavy data augmentation is used such as random cropping, angle rotations, horizontal and vertical flipping, exposure and saturation. We turned off hue augmentation as the mammograms are grayscale images. We used the input image scale of 416 x 416. Following are some of the hyperparameters used for while training:

- Batch size = 64
- Number of minibatches = 16
- Minibatch size = 64/16 = 4
- Momentum = 0.9
- Decay = 0.0005
- Angle augmentation = -10 degrees to +10 degrees
- Saturation scale = 1.5
- Exposure scale = 1.5
- Hue scale = 0
- Learning rate = 0.001

We calculated the anchor boxes specific to this dataset using the K-means clustering algorithm.

Training using RSNA, DDSM and AIIMS datasets

Owing to the large network and the number of trainable parameters, as well as the fact that the MAP scores of the above procedure indicate overfitting. Hence we decided to use transfer learning using a chest X-Ray dataset as proposed by the paper Transfer Learning from Chest X-Ray Pre-trained Convolutional Neural Network for Learning Mammogram Data Neural Network for Learning Mammogram Data.
We kept the network unfreezed due to large size of the RSNA dataset, which contains 30000 chest X-ray scans. We trained the network for 6k iterations and stopped training when the loss converged.

We then finetuned this network using the DDSM dataset and added normal images to the dataset unlike earlier when we trained the model using DDSM + AIIMS datasets. This is due to the fact that the author of the repo indicated to use 50% normal images to increase the MAP

score and optimise training. We finetuned this net by freezing the first 81 layers and kept the rest layers unfreezed, ie, around 5 million free parameters. We later even included the normal images into our training set to indicate to the network what not to pick up.

Once this network was finetuned, we used this network to finetune using the same freezing procedure with the AIIMS dataset. This is due to the fact that AIIMS dataset is visually similar to our testing dataset, INbreast and hence it makes sense to finetune the layers to learn similar features as INbreast.

Training using DDSM, INbreast and AIIMS datasets

Here the network was trained on whole of DDSM, whole INbreast along with a part of the AIIMS dataset. It was tested on the remaining AIIMS data.

RetinaNet

Training using DDSM and AIIMS datasets

RetinaNet was trained on the combination of DDSM and AIIMS including normal images which nearly added up to 4500 cases (either benign lesion, or malignant lesion or normal).

- The models was trained for 100 epochs
- The models formed within the first 20 epochs made more accurate predictions
- Batch size was set to 2 and number of workers 3
- Adaptive Learning Rate was used
- Learning rate was initially set to 1e-5
- The basenet to extract features was ResNet 50
- The ResNet-50 was pretrained on ImageNet Dataset
- Images are being resized to 608 x 1024 along with normalization
- In-built data augmentation of image flipping is being performed
- Gamma value was set to 2
- Adam optimizer was used

The predictions on INbreast were accurate, however, the models beyond 20 epochs did not perform well which could be an indication of overfitting.

Training using DDSM, INbreast and AIIMS datasets

Here the network was trained on whole of DDSM, whole INbreast along with a part of the AIIMS dataset. It was tested on the remaining AIIMS data. We also analyzed the relative performance on masses and calification.

Training for Actionable vs Non-actionable

We also performed experiments by relabelling the classes as actionable and non-actionable since it is often hard to distinguish between BIRADS 3 and BIRADS 4a and sometimes BIRADS 2 which hence leads to radiologist bias and inconsistencies. Actionable vs non-actionable categorization solves that problem.

3.2.6 Proposed Network 1

Since a single model indicated a strong ability to detect masses but were not sufficient for detection of calcifications which are much smaller and very different as compared to masses. We hence concluded that the same model cannot generalize to detect both masses and calcification. To address this problem, we developed an expert - discriminator network which trains two experts - one for masses and the other for calcifications. Both the experts were trained using the same faster RCNN framework, the only difference was that the loss terms were considered only for masses in the first expert and only for calcifications in the second. The aim was to overfit the one on masses and the other on calcifications so that the two units together are able to detect both types of lesions with high accuracy.

Post training, when a new image is to be processed, it is sent through the two experts which will both perform their detections. The max scores from the two models is picked and those predictions are made.

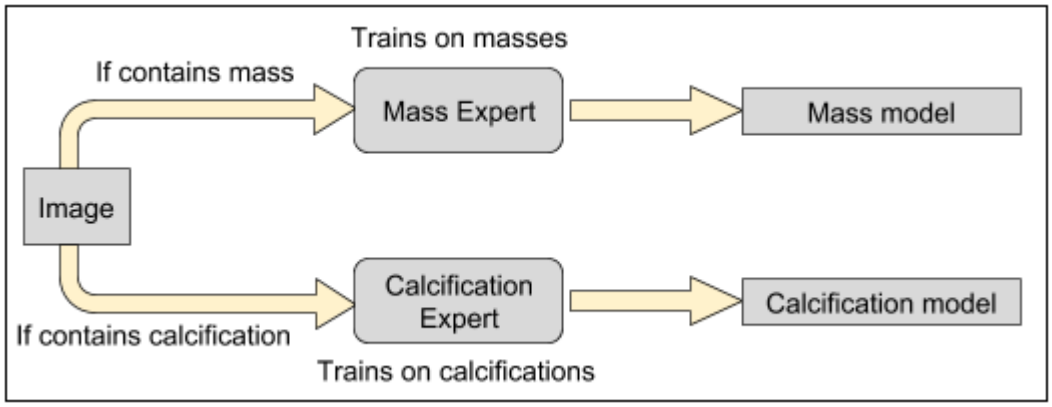


Figure 3.10: Proposed 2 Expert Model

3.3 Current Work

3.3.1 Experiments

A range of experiments were conducted to evaluate the classification and detection performance of the network. These include mass vs calcification analysis, actionable vs non-actionable training and testing, training on small masses, testing on a set of 20000 class labelled mammograms and obtain results.

Mass vs Calcification Performance

The model trained on set of images from DDSM and INbreast and a fraction of the AIIMS data was tested on remaining AIIMS data. Evaluation was done on not only by studying benign and malignant correctness but also by analyzing how the network performs independently on masses and calcification. The same was also done by training the network on DDSM, AIIMS and testing on INbreast. The following values were calculated:

- Number of malignant lesions
- Number of malignant masses
- Number of malignant calcification
- Number of malignant lesions correctly predicted as malignant
- Number of malignant masses correctly predicted as malignant
- Number of malignant calcification correctly predicted as malignant
- Number of malignant lesions not predicted

Actionable vs Non-actionable

An alternative annotation scheme was decided since it is often hard to evaluate between subtle differences in BIRADS 2, 3 and 4a. In addition, it often a radiologist-dependent. To get rid of such biases, also since our labels were derives from annotations of a single radiologist without validation or confirmation from any other radiologist, we relabelled the lesions in a way to categorize them into Actionable and Non-actionable. We clubbed BIRADS 1 and 2 into Non-actionable and BIRADS 0, 3, 4 and 5 into Actionable, re-trained the network and on DDSM, AIIMS and tested it on INbreast.

Small Masses

A small set of 77 malignant AIIMS masses were annotated to study the performance of the networks on masses which are not relatively small. The network is often able to locate masses

of a considerable size, however, performance on smaller sized masses had to be checked. The model trained on DDSM and Inbreast was tested on these samples.

Image labelled set

In addition to datasets with annotated lesions, since a detection network has classification capability as well, we used the RetinaNet model to test on a set of 20,000 AIIMS images was only image level labelling - Normal, Benign or Malignant. This dataset was put together by other members working on the same project. The dicom images were read, label was extracted and infused into the image name. Results indicated that RetinaNet could produce good performance on classification as well.

Besides these experiments, we tweaked our testing strategy by changing the scoring metrics, for example, consider concentric predictions as False positive instead of True Positive, or completely ignoring. We also varied the thresholds for FROC and obtained visualizations to study the missed out lesions.

3.3.2 Sanity Check

An important consideration to make while evaluating the obtained results is to ensure they are obtained as expected and are thus consistent with out understanding of the network's working. For this, we needed methods to ensure that the important information in the image which is yielding the predictions is indeed coming the same area of the image which has the object and not from some other unrelated part in the image. For this, we need to check the attention maps produced by the network and analyze the salient regions. For this we attempted several different techniques - Making CAMS, GRADCAMs, performing sliding window analysis and perturbation analysis.

CAM

Convolution layers do have localization properties but they are lost due to fully connected layers for classification. Global average pooling (GAP) can be used for localization in an image as it can determine the global extent of objects in the image. A class activation map for a particular category indicates the image regions used by the Convolutional Neural Network. GAP layer is added just before final output layer. Class activation mapping is generated which is basically projecting the weights of the output layer on to the convolutional feature maps. GAP outputs the spatial average of the feature map of each unit at the last convolutional layer. Every output obtained from GAP of channel k is weighted with weights w(k,c). The weights w(k,c) indicate the importance of feature map k for class c. A weighted sum of these these feature maps is used to generate the final output $\bar{S}(c)$. Softmax is evaluated over all S(c) values and the desired class c_chosen is selected from this (say max). Now the CAM can be obtained by upsampling the

weighted combination of feature map f(x,y,k) and w(k, c_chosen) to the size of the input image. Image regions and objects corresponding to c_chosen can be visualized.

GRADCAM

Compared to CAM, GradCAM doesn't require feature maps to be directly before the softmax layers like in the case of CAM. Due to this reason, CAM is only applicable for specific types of Neural Nets while GradCAM are a generalisation to CAM and can be used with a variety of CNNs. The algorithm steps are briefly as follows: Gradients of the score(before the softmax layer) for the chosen class C are computed: yc with respect to feature map Ak of the last conv layer. Gradients are global average pooled (mean over all the gradients calculated for every position of the map) to get weights Bk. This weight Bk captures the importance of the feature map k for a target class c. Weighted sum of the feature maps (where the weight for the kth feature map is Bk calculated in the previous step). Finally this map is passed through the Relu function to remove the negative pixels as they belong to the classes not in consideration.

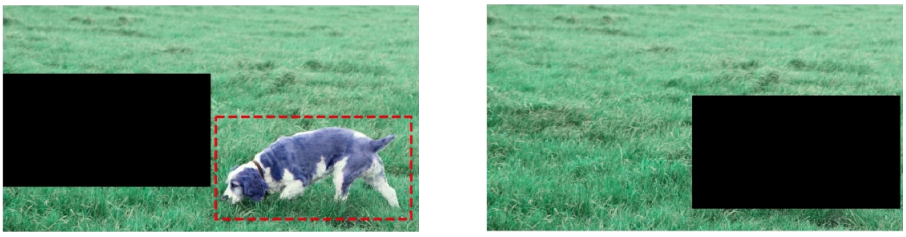
We tried to implement the GradCAM for Faster-RCNN visualization. However, the attention regions were not consistent with where the object was in the image. We could also not find any other resources where GradCAM has been implemented for Faster-RCNN to validate our steps. We suspect the reason for inconsistent CAMs could be the intermediate Region Proposal Network (RPN), which could be leading to incorrect gradient calculation. The gradient is calculated for the score for a class with respect to the feature map of the last conv layer, but here since they are separated by the RPN, we are unsure if it can yield the same performance as in a classification network.

Since our aim is to verify that Faster-RCNN does have localization capabilities, we have tried 3 other different methods for the same using occluded regions - where we blacken our parts of the input image and analyze the changes. Our hypothesis is that if blackening out a certain region of the image is causing major changes in the predicted boxes/confidences, then that region was salient and important for the task.

Sliding Window Analysis

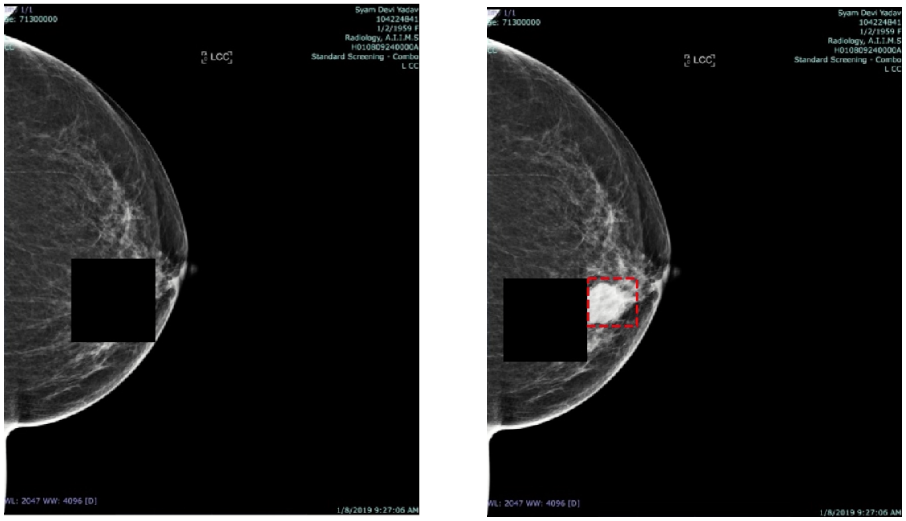
For the sliding window analysis, the image was occluded in sequence at different sliding window positions. The occluded boxes were chosen in three different scales and placed at every position in the image. The occluded images with predictions were played in the form of a video to understand clearly the changes in the predictions as the occluded block moves across the image regions. The hypothesis is that if the predicted boxes are significantly changes on occluding a certain region of the image then that region is important and contributed to the predictions highly thus a salient region with high attention. This technique was applied to both normal Imagenet images and to breast mammograms. It was made evident that the region of

importance was solely the region containing the objects - cat/dog for Imagenet and the lesion - mass/calification for the breast mammogram. This thus reiterated that the finding are relevant and the network performs as expected.



(a) (b)

Figure 3.11: Two types of Mammograms



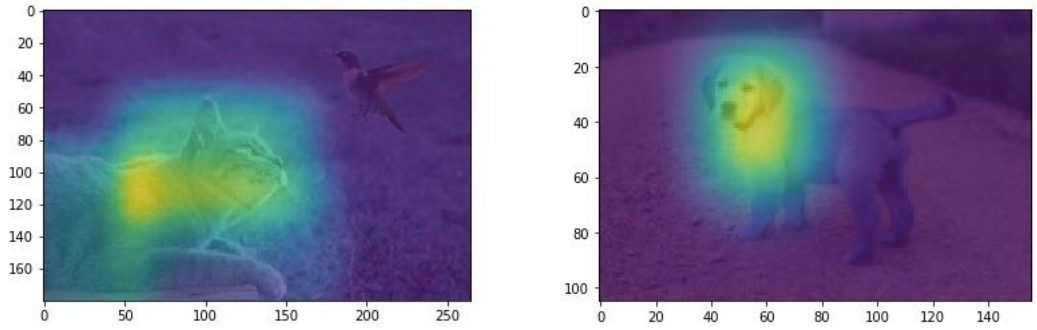
(a) (b)

Figure 3.12: Two types of Mammograms

In addition to this Sliding Window approach, We developed a novel alternative technique to generate attention maps using perturbations. The two variationg of the algorithm are discussed in detail below.

Perturbation Technique - I

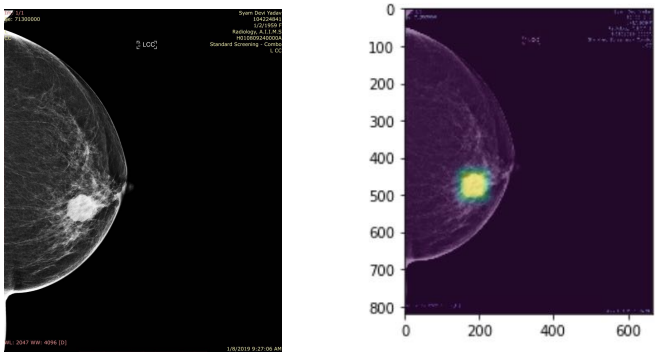
Take an image and make occlusions on the image in a sliding window style where the window is of 3 scales or sizes, approximately 1000-2000 images per test image depending on the size of the image. Run these images through the Faster-RCNN model. First did this for normal images like cats and dogs and ran it through FRCNN trained on VOC. Once we get the predictions for these images, for each pixel in the image we calculate a score. This score is the sum of the confidence scores of the bounding boxes in which the pixel is present. Then we normalize these scores and make a heat map. We got the following heatmaps for cats and dogs:



(a) (b)

Figure 3.13: Saliency maps for a) cat and b) dog

The same procedure was tried for a breast image and the pretrained FRCNN model.

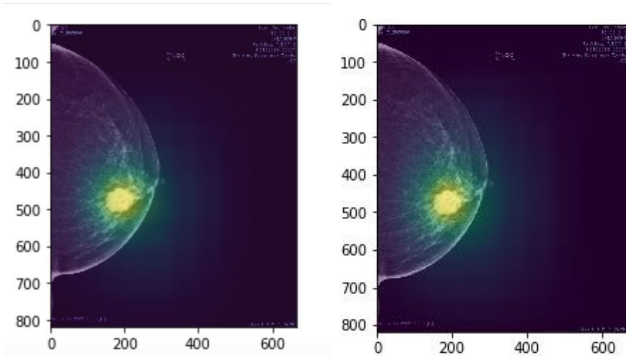


(a) (b)

Figure 3.14: a)original image and b)the corresponding saliency map

Perturbation Technique - II

The above procedure has certain flaws which are eradicated in the perturbation technique II: The same procedure as above is followed initially; do the occlusions and get the predictions of these occluded images. Then we output the predictions of the original non-occluded images using FRCNN pretrained model. Then for each occluded image, we calculate the IOU between the predicted bounding box of that image and the predicted bounding box of the original image. We then add value : 1-IOU, to all the pixels within the occluded region. The reason for doing this is that if the IOU is high, that means that the occlusion didnt result in any change, thus the occluded region didnt have any attention. Whereas on the other hand if the IOU is low, that means that occlusion affected the prediction and hence is comprised of the attention region. If for a given occluded image there is no prediction, IOU =0. Finally we take this heatmap and normalize it and super-impose on the original image. Here are the results for normal images(using model trained on VOC) and breast images, using pretrained FRCNN model and Retinanet:



(a) (b)

Figure 3.15: Saliency maps for a) breast for FRCNN model b) breast for Retinanet model

3.3.3 Proposed Network 2

After some analysis of the false positives and false negatives from the various testing datasets we used, we found out that in most of the cases the misses were in breasts which were dense. After performing experiments on mass vs calcification performance we realised that the shortcoming is not in separately identifying mass and calcifications, but in lesions belonging to breasts of different densities. Particularly, it was noticed that obscure masses often hidden in the fibro-grandular tissure were undetected. To tackle this problem, we hypothesized that if the dense tissues are normalised using an intensity transformation, the results may improve. Therefore we designed a network for density contrast modification. Our hypothesis was backed by the results presented in the paper: Photometric Transformer Networks and Label Adjustment for Breast Density Prediction.

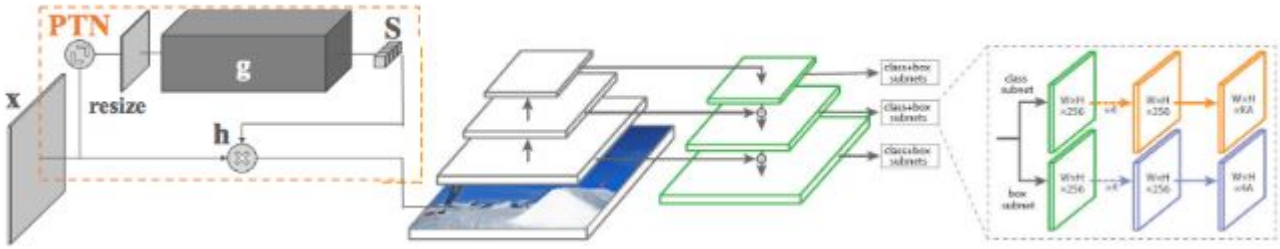


Figure 3.16: Proposed Network 2

We use Photometric Tranformer Network to normalise the intensity of the images and hence reduce the deviation in the density of the breasts. For doing so, we designed a 5 layered Convolutional Neural Network with a 10 softmax outputs (S). These are the k=10 parameters which will be used in the function 'h' to transform the input images. The authors introduce a function h, which uses k parameters. These k parameters and the function h is used to transform the input image and normalise it's density. This neural network is trained on the fly with the Retinanet. As the function h is continuous and differentiable at all points, the loss can be fully back-propagated.

Chapter 4

Results

4.1 Baseline

4.1.1 On InBreast

Faster RCNN

ROC AUC = 0.941

FROC: Sensitivity at 0.3 False positive per image = 0.88

YOLOv3

ROC AUC = 0.85

FROC: Sensitivity at 0.3 False positive per image = 0.90

RetinaNet

ROC AUC = 0.946

FROC: Sensitivity at 0.3 False positive per image = 0.90

4.1.2 On AIIMS

Faster RCNN

ROC AUC = 0.805

FROC: Sensitivity at 0.3 False positive per image = 0.68

RetinaNet

ROC AUC = 0.847

FROC: Sensitivity at 0.3 False positive per image = 0.857

4.2 Mass vs Calcification

4.2.1 On AIIMS

Total malignant lesions = 138; 104 masses and 34 calcifications

Faster RCNN

90 out 138 detected, 48 missed

68 are masses and 22 are calcifications

RetinaNet

111 out 138 detected, 27 missed

86 are masses and 25 are calcifications

4.2.2 On INBreast

Total malignant lesions = 91; 76 masses and 14 calcifications

Faster RCNN

80 out 91 detected, 11 missed

67 are masses and 13 are calcifications

RetinaNet

78 out 91 detected, 13 missed

67 are masses and 11 are calcifications

4.3 Small Masses

4.3.1 Faster RCNN

ROC AUC = 0.993

FROC: Sensitivity at 0.3 False positive per image = 0.86

4.3.2 RetinaNet

ROC AUC = 0.993

FROC: Sensitivity at 0.3 False positive per image = 0.89

4.4 Image-labelled 20,000

4.4.1 Faster RCNN

ROC AUC = 0.818

4.4.2 RetinaNet

ROC AUC = 0.821

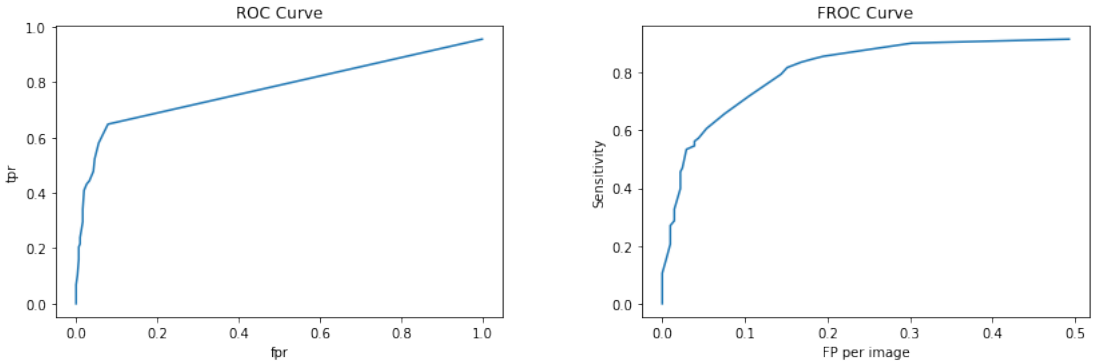
4.5 Photometric Tranform Network

4.5.1 RetinaNet

ROC AUC = 0.78

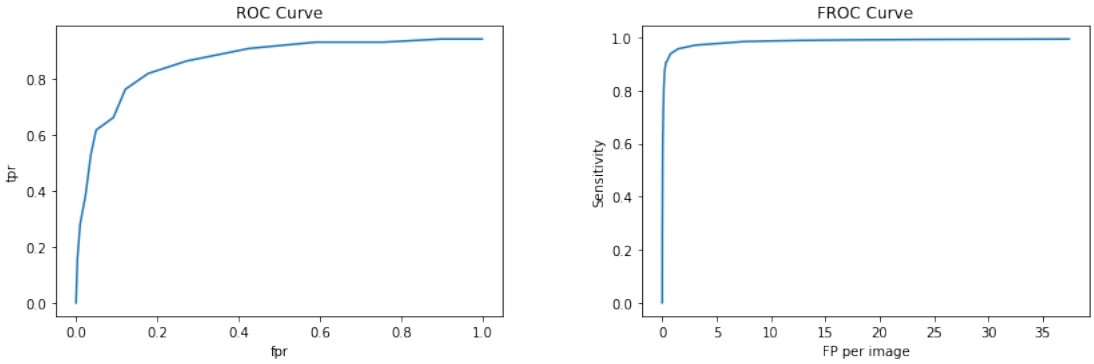
We are currently doing more analysis on using Photometric Transformer Network with RetinaNet and Faster RCNN. We are in the process of adding Hinge Loss term as a counter for the negative pixel values which may result score improvements.

4.6 Plots



(a) ROC (b) FROC

Figure 4.1: ROC and FROC for YOLOv3



(a) ROC (b) FROC

Figure 4.2: ROC and FROC for RetineNet

4.7 Visualisation

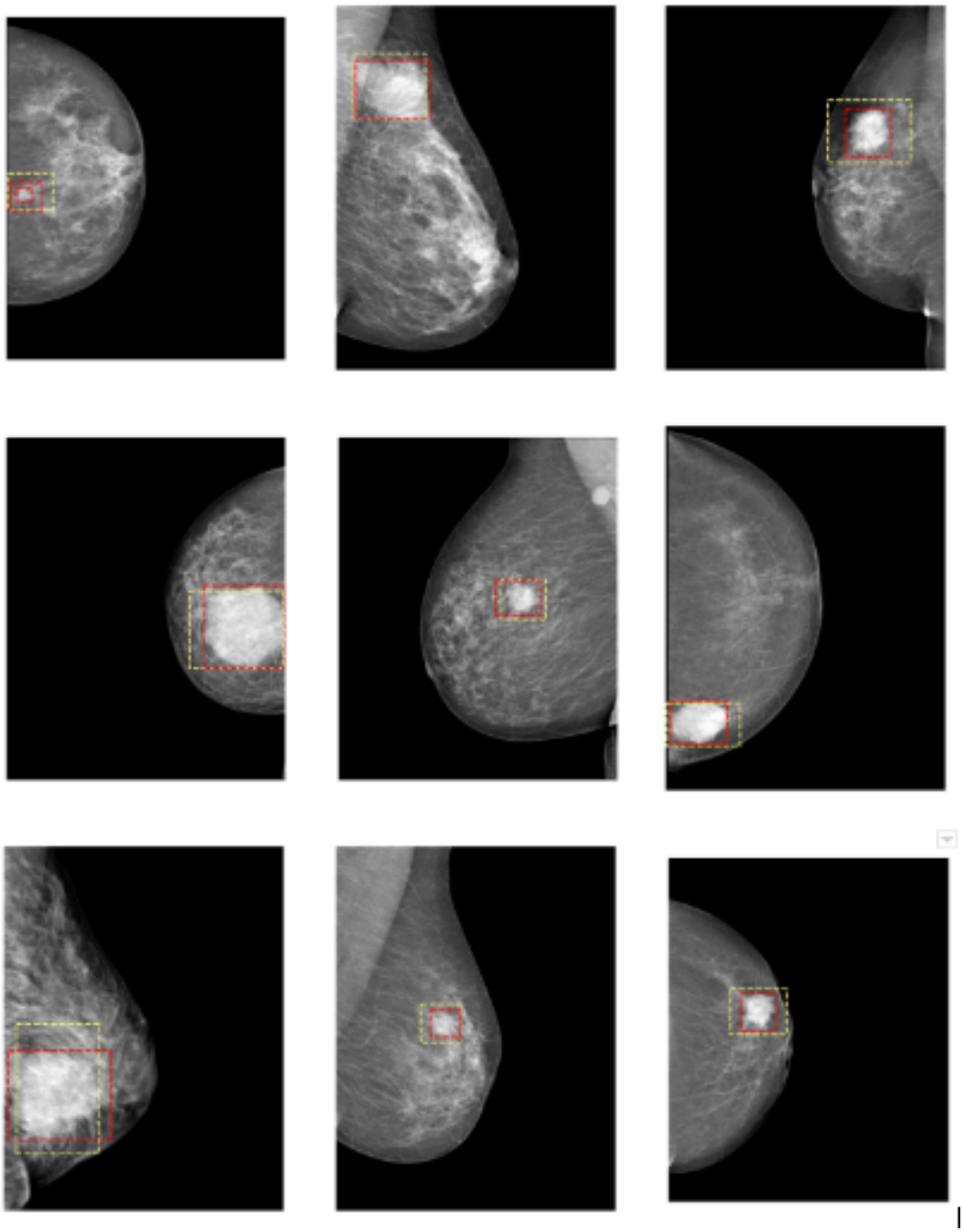


Figure 4.3: Visualizations of Detections for YOLOv3

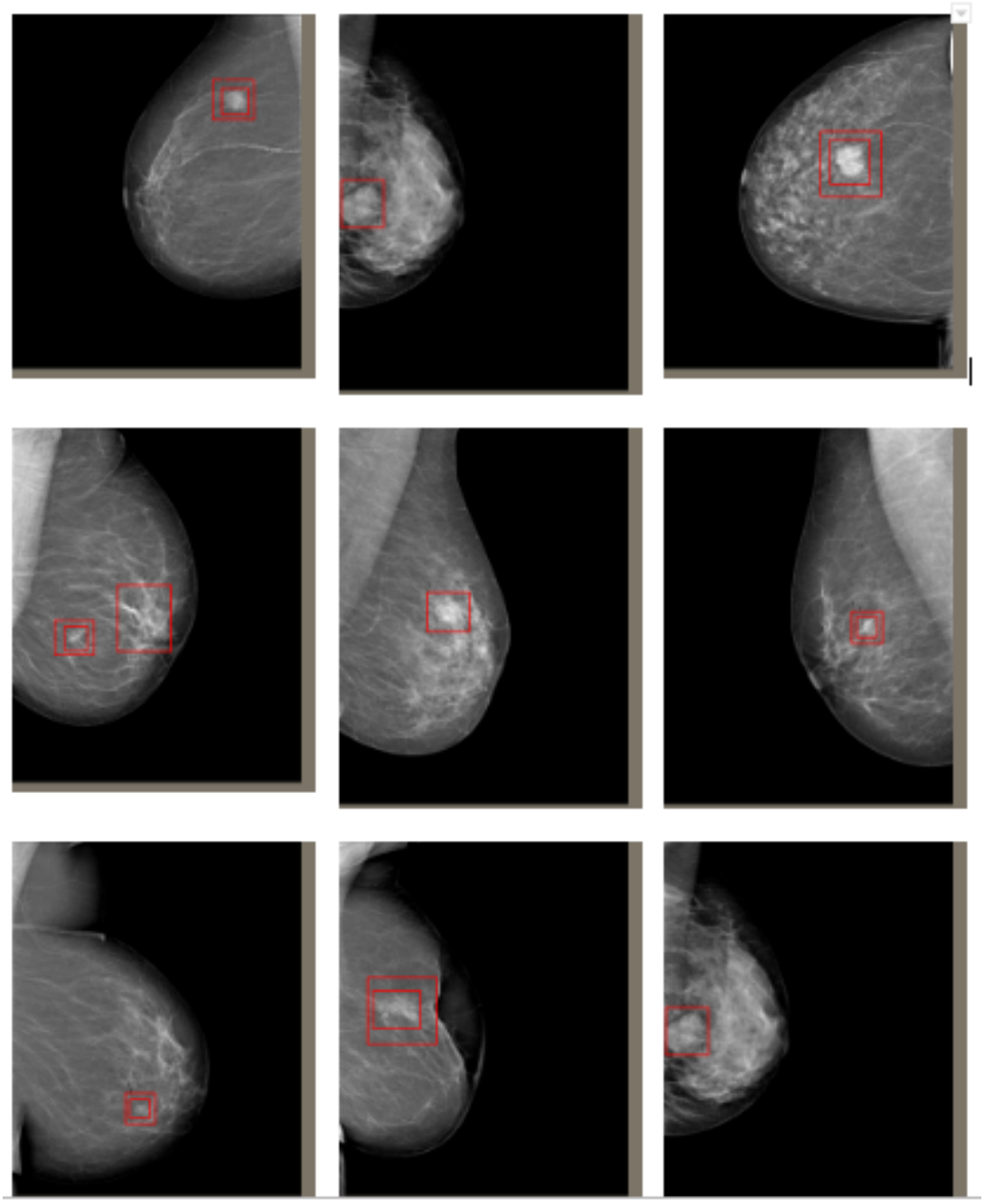


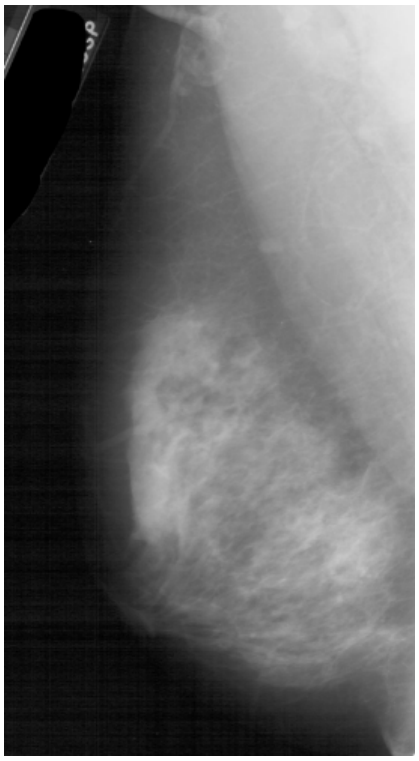
Figure 4.4: Visualizations of Detections for RetinaNet

Chapter 5

Challenges

We faced several challenges during the entire course of our thesis work:

- Gaining the domain knowledge was very important for us to be able to train our models well and also understand the datasets etc. Gaining the required background knowledge took a long time though we thoroughly enjoyed it.
- Setting up the Caffe framework on the machine was a very tedious process since it comes with several dependencies and every software has different versions. For the entire system to work, all versions (OpenCV, CUDA, Caffe) had to be compatible, hence we took nearly one and a half months to setup the system.
- Since we used different datasets, they all came in different formats. Hence, along with understanding the datasets, and preprocessing the images, we also had to convert them to a suitable format.
- Annotation formats were different for different datasets. We had to convert them to xml files in PASCAL VOC format. Though these datasets are very popular, we did not find any resources to help us in this task, hence it took us a considerable amount of time.
- One of the most popular breast mammogram dataset – DDSM provides the maximum number of images. However, these images were in a lossless compressed ljpeg form and reconverting it to jpeg was a major challenge since these images had different bit depths. Also, DDSM images were not of best quality and contains a few inconsistent annotations as well.
- Two varieties of mammograms exist – screen film and digital. These two are very different and DDSM being screen film, testing on INBreast, a digital database was a challenge.
- Due to the very high complexity of object detection networks and millions of parameters, controlling the network from overfitting was a major challenge. In order to tackle this, we had to apply various freezing, augmentation and dropout technique which required heavy experimentation.



(a) Screen Film Mammogram

(b) Digital Mammogram

Figure 5.1: Two types of Mammograms

- Performing sanity checks for every result was extremely important. Most literature do not provide convincing methods that guarantee the correctness and working of the system. We had to device several new strategies to conduct such checks, which was both painstaking and challenging.

Chapter 6

Conclusions and Future work

We matched the state of the art ROC classification and FROC localization scores using 2 separate architectures: YOLO V3 and RetinaNet. Following, we proposed 2 of our own networks - the Ensemble Expert Network and the Photometric Transform Network, aimed to tackle the limitations of previous works by ensuring systematic capturing of lesions of different type/texture and incorporate variations in breast density to enable easier detection of obscure masses. In addition, we developed a new stragy for attention visualization - Perturbation Analysis, which is a viable alternative to CAM/GRADCAM and can apply to detection networks as well. We even utilized different nomenclature and class naming to categorize the data differently to improve results. We tested on several sets of data obtained from AIIMS including a set of small masses, a set of mass and calcification labelled images, a set of 20,000 image-levelled labelled images and on INbreast data. The results are promising, show significant improvements in performance and demostrate the importance of proper dataset, annotations, a suitable architecture and network design and testing techniques. Currently, we are working on improvements to the Photometric Network. Adding a hinge loss term can enable removal of negative pixel values, which is what we are trying to incorporate in the loss function of the network. Moreover, altering the several available parameters like u, v, and k can lead to great differences in constrast enhancements. Next, we plan to replace the resnet backbone with the SE-resnet tor Squeeze and Excitation Network to fuse spatial and channel information, which could lead to improved scores.

Bibliography

[1] Nature paper on Breast Cancer detection
https://www.nature.com/articles/s41598-018-22437-z

[2] Paper Implementation
https://github.com/riblidezso/frcnn_cad

[3] Faster-RCNN,
https://arxiv.org/abs/1506.01497

[4] YOLO V3,
https://pjreddie.com/media/files/papers/YOLOv3.pdf

[5] YOLO V3 Implementation,
https://github.com/pjreddie/darknet

[6] YOLO V3 Implementation,
https://github.com/AlexeyAB

[7] YOLO Paper,
https://www.ncbi.nlm.nih.gov/pubmed/29477437

[8] YOLO Paper,
https://www.researchgate.net/journal/1386-5056_$International_Journal_of_Medical_Informatics$

[9] RetinaNet,
https://arxiv.org/abs/1708.02002

[10] RetinaNet Implementation,
https://github.com/yhenon/pytorch-retinanet

[11] Architecture images from,
https://medium.com/@smallfishbigsea/faster-r-cnn-explained-864d4fb7e3f8

[12] Caffe arhitecture visulisation,
http://ethereon.github.io/netscope/quickstart.html

[13] Caffe framework,
http://caffe.berkeleyvision.org

[14] Parallel Convolutional Networks for Image Recognition via a Discriminator,
https://arxiv.org/abs/1807.02265

[15] Transfer Learning from Chest X-Ray Pre-trained Convolutional Neural Network for Learning Mammogram Data Neural Network for Learning Mammogram Data,
https://www.sciencedirect.com/science/article/pii/S1877050918314807

[16] Learning Deep Features for Discriminative Localization,
http://cnnlocalization.csail.mit.edu/Zhou_Learning_Deep$_F$*eatures_CVPR_*2016*_paper.pdf*

[17] Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization
https://arxiv.org/pdf/1610.02391.pdf

[18] Photometric Transformer Networks and Label Adjustment for Breast Density Prediction
https://arxiv.org/pdf/1905.02906.pdf