Improving Breast Cancer Screening Exam Classification Using Localization Based Approach

by

Sudarshini Tyagi

advised by

Krzysztof Jerzy Geras

read by

Kyunghyun Cho

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Masters of Science
Computer Science
New York University

May, 2020

Abstract

Breast cancer is the second leading cause of cancer-related deaths among women in the United States. Screening mammography has been highly effective in reducing mortality rate but suffers from a high rate of false positive recalls, leading to added costs and stress. Better computer-aided diagnostic tools could improve patient outcomes by helping radiologists. Our goal is to build a system that provides highly accurate as well as interpretable predictions, in order to assist radiologists in cancer detection. To this end, we train an object detection network to predict the location of suspicious lesions and classify them. Our model achieves an AUC of 0.9088 without ensemble in predicting malignancy in patients undergoing breast cancer screening. In addition, the model generates attention weights highlighting the areas that contribute to benign and malignant findings, providing interpretable predictions.

The aim of this thesis is to create a self sufficient text that covers all the previous work in this area and also describes the components in great detail while covering the new improvements and experiments tried by us.

Contents

Listing of figures

Acknowledgments

Of the many people who have contributed to this work, I would like to first of all thank my supervisor, Krzysztof Jerzy Geras. His determination to this field, his understanding of the challenges of this domain and his constant support made working with him an amazing learning experience. I am also fortunate to have been part of the research group he is leading. I would also like to thank my second reader Kyunghyun Cho for his helpful input. I would particularly like to thank fellow students Nan Wu, Artie Shen and Ashwin Bhola for their helpful comments on this work and its precursors. This work was supported in part by grants from the National Institutes of Health (R21CA225175 and P41EB017183).

0

Introduction

0.1 Problem Statement

Breast cancer is the second leading cause of cancer-related deaths among women in the United States. Screening mammography has been highly effective in reducing mortality

rate but suffers from a high rate of false positive recalls, leading to added costs and stress. Better computer-aided diagnostic tools could improve patient outcomes by helping radiologists.

0.2 Important Definitions

0.2.1 Breast Cancer Screening

Best Cancer Screening are a set of tests that are performed before there are any warning signs or symptoms. The tests include mammography and clinical breast exam. For some women at higher than average risk of breast cancer, breast MRI may also be used.

0.2.2 Biopsy

Biopsy is a procedure that is performed if the screening tests show something suspicious. The breast biopsy is a test that removes tissue or fluid from the suspicious area which are then examined under a microscope and further tested to check for the presence of breast cancer. It is the only diagnostic procedure that can definitively determine if the suspicious area is cancerous.

0.2.3 Mammograms

A mammogram is an X-ray picture of the breast. X-rays use invisible electromagnetic energy beams to produce images of internal tissues, bones, and organs on film. A mammo-

gram is used to detect the cancer in it's early stages as it can be three years before it can be felt. Using a mammogram, it is possible to detect a tumor that cannot be felt. Mammography cannot prove that an abnormal area is cancer, but if it raises a significant suspicion of cancer, tissue will be removed for a further biopsy. In this work, the mammograms form the dataset for the model.

0.2.4 Tumor

A tumor is an abnormal lump or growth of cells.

0.2.5 Benign Tumor

When the cells in the tumor are normal, it is benign. Something just went wrong, and they overgrew and produced a lump. It won't invade nearby tissues or spread to other areas of the body.

0.2.6 Malignant Tumor

When the cells are abnormal and can grow uncontrollably, they are cancerous cells, and the tumor is malignant. Malignant means that the tumor is made of cancer cells, and it can invade nearby tissues. Some cancer cells can move into the bloodstream or lymph nodes, where they can spread to other tissues within the body(metastasis).

0.3 Current Challenges

Breast Cancer screening using X-ray is the most popular and effective method to identify the cancer that are occult or not showing symptoms. However, they still suffer from the problem of too many false positives recalls which can be very expensive and can also lead to anxiety in the patient. Currently, there is a recall rate of 10-15% following the screening mammography which equates to about 3.3 to 4.5 million callback exams. Hence, there is certainly a need to make the process of breast screening more effective and seamless.

One way to do this is to provide the radiologist with a tool that can give them useful insights about the X-ray image to assist them through the process. Recently, deep learning has seen a huge success in tasks like image classification and object detection in natural images. Hence, there is a huge promise in applying those techniques to medical images for automatic detection and classification. However, medical images differ from natural images as the former are of much higher resolution than the latter that have been used in the various successful deep learning tasks. One challenge that arises because of this is that our task becomes very computation heavy. Down-scaling to a lower resolution can work in natural images as it is computationally efficient and as such there is no significant improvement when using a higher resolution image. This is because of the property of natural images to present the objects of interest in relatively larger portions than other objects and hence only the macro objects matter.[2] This is however not the case in medical images as the areas of interest are often subtle and do not represent a large portion of the image especially in an early stage cancer. Hence, down-sampling negatively impacts the performance of the model. Our approach will takes all these concerns into consideration while designing and

conducting our experiments.

0.4 Our Approach and contributions

Our goal is to build a system that provides highly accurate as well as interpretable predictions, in order to assist radiologists in cancer detection. To this end, we train an object detection network to predict the location of suspicious lesions and classify them.

The day healthcare can fully embrace AI is the day we have a revolution in terms of cutting costs and improving care.

Fei-Fei Li

1

Background and Related Work

This section contains a very brief introduction to the concepts of machine learning and deep learning. It further expands on the concept of object detection and the various algorithms for the task. The final section discusses the related work to this thesis. This includes all the previous work that this project builds upon and also the parallel work in breast can-

cer detection.

1.1 Machine Learning

Machine Learning is a sub-field of artificial intelligence wherein a machine learns to do a given task automatically through the experience. The algorithms typically try to create a mathematical model based on sample data in order to make certain decisions without explicitly programming the machine to do it.

Machine Learning can be supervised or unsupervised. If the algorithm is given a set of inputs $\{x_1, x_2, .., x_n\}$ and a set of corresponding outputs $\{y_1, y_2, .., y_n\}$ and the goal of the algorithm is to learn to predict the right output for a given new input, then the task is called supervised as we are providing supervision in the form of outputs or labels. However, if the algorithm is only given a set of inputs $\{x_1, x_2, .., x_n\}$ then the task is called unsupervised. The task that we are working with here is a supervised task.

1.2 Deep Learning

Deep Learning is a sub-family of machine learning algorithms. These methods comprise of artificial neural networks and also rely on representation learning. Neural networks and deep learning currently provide the best solutions to many problems in image recognition, speech recognition, audio processing, and natural language processing. Conventional Machine Learning algorithms are limited in their capacity to process the raw data. This is because for a long time, machine learning systems required carefully engineered features

extractors which was not possible without considerable domain expertise. These feature extractors would transform the raw data into suitable internal representation or features that could be fed into the machine-learning algorithm as input. However, representation learning allows the machine to automatically discover these features or internal representations needed for the task. Deep learning methods heavily rely on representation learning methods.[8]

A typical Deep Learning setup consists of 1. A multi-layer neural network, 2. A loss function, 3. A back-propagation algorithm. The network takes the input space and distorts it to make the classes of data linearly separable. There was various ways to distort the space, hence, to find the right distortions, the network starts with a set of initial parameters called weights, computes the final class of the input and compares it to the actual output for that image, called the label. Now, it calculates the loss which signifies how far is the network's output from the actual output. This loss is then back-propagated to the network using a back-propagation algorithm and the network's weights adjust themselves iteratively to match the actual output or until the loss reaches a minimum. The multiple layers of the network learn hierarchical representations or features that are fed to the final layer that acts as a classifier. Hence, the deep learning setup can learn the features on it's own without any extra engineering or domain expertise.

1.3 CNNs

Deep convolutional networks are models that are designed to process data that comes in multiple arrays, consider for example a color image that is made up of three 2D matrices.

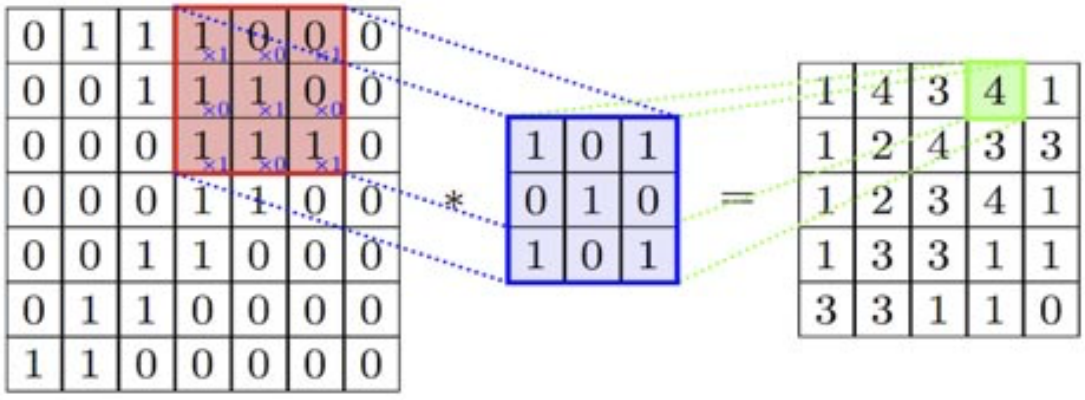

Figure 1.1: 2D Convolution Operation

The three matrices represent the three channels of the image Red, Blue and Green. This makes them ideal from image processing tasks like image classification, detection, video processing, etc. Conv nets are based on four key ideas: 1. local connections, 2. shared weights, 3. pooling, and 4. use of many layers.[8] These ideas take advantage of the properties of natural images. In images, local group of pixel values are often highly correlated that form motifs that are easily detected. Also, this is invariant of location, meaning, a motif can appear in any part of the image. This forms the basis of convolutional units.

Consider an image with 3 channels R, G, B. Let the resolution (H X W) of the image be 28 x 28. Now, we have a volume of 28 x 28 x 3. A kernel or a convolutional unit is a matrix of size H_k x W_k x depth of input features(3 here) which is learned automatically using the deep learning paradigm explained in the previous section. Figure 1.1 explains how the convolution works in a 2D setting. When we have a 3D kernel as in this case, the entire volume of the kernel is multiplied point-wise with the feature volume and the elements are then added to form one single number. This result is then passed through a non-linearity such as ReLU.

One convolution layer consists of many such filters. When an input image of dimensions H x W x C is passed to the network, and if the first conv layer is defined as H_k x W_k x $num - filters$ then the output of this layer would be H_{new} x W_{new} x $num - filters$ where,

$$H_{new} = \frac{H - H_k + 2P}{S} + 1$$

Here, S = Stride is the number of pixels by which the kernel is shifted every stride and P = Padding is the number of pixels by which we pad the input on all sides.

Similarly,

$$W_{new} = \frac{W - W_k + 2P}{S} + 1$$

After the convolution, the pooling layer is applied. While, the role of convolution layer is to detect the local conjunction of features from the previous layers, pooling layer merges the semantically similar features.[8] This combination of convolution, non-linearity and pooling forms a block. Few blocks are repeated or stacked to compute various hierarchical representations and finally fully connected layers are added as classifiers. Backpropagation is similar to the regular deep learning networks and the weights of the kernels are learnt are learnt.

1.4 Object Detection

1.4.1 What is Object Detection?

Object Detection is a computer vision task that deals with identifying and locating objects of certain classes in the image. Object localisation can be done in various ways. One method is to draw bounding boxes around the object and another method is to mark every pixel of the image that contains the object called segmentation.

Figure 1.2: Object Detection using bounding boxes

Figure 1.3: Object Segmentation

1.4.2 Types of object detection systems

Object Detectors can largely be classified into two categories namely traditional object detectors and deep learning object detectors. The traditional detectors were mainly developed before the advent of the successes of deep learning. These include Viola Jones Detectors, HOG Detector, Deformable Part-based Model (DPM), etc. Most of these algorithms were built on hand crafted features as there was a lack of effective image representation learning at the time (before 2014). However, as the performance of hand crafted features plateaued, not much improvement was seen and the only gain was from minor variants or ensemble systems.[24]

In 2012, convolutional neural nets had a breakthrough and with that object detection saw the promise. Indeed, today we have deep learning based object detectors which are much better in performance and much easier to train than their traditional counterparts. In this text, we will expand more on these types of models as we are also using one of the variants for our work. They can be further divided into two types: 1. Two stage detectors, and 2. One Stage Detectors

Two stage detectors

In 2014, R-CNN (Regions with CNN Features)[3] was proposed by R. Girshick et al. which became the first object detector using deep learning. R-CNN is a two stage detector. The two stages involve, 1. Extracting a set of object proposals(candidate bounding boxes) using selective search, and 2. Rescale these proposals to a fixed size image, feed it to the CNN,

extract features which are finally fed to linear SVM(Support Vector Machines) classifiers. These classifiers then detect the presence or absence of the object and if present, they also assign a class to the object. R-CNN showed a significant boost in performance compared to traditional detectors. However, R-CNN had a few drawbacks. First, training happens in multiple stages which makes the whole process slow. Second, for SVM and bounding box regressor training, features are extracted from each object proposal in each image separately without sharing computation. This leads to a lot of redundant computation as most of these potential bounding boxes are overlapping. This requires a lot of storage and time.

Spatial Pyramid Pooling Networks (SPPNets)[5] solved the problem of redundant computation by proposing shared computation. It computes a feature map for the entire image and then features of the proposals are extracted from this feature map to classify the individual proposals. SPPNet was approximately 20 times faster than R-CNN without any loss of performance. However, it was still a multi-stage detector involving feature extractors, fine tuning the network, traning SVMs and fitting bounding box regressors.

In the next year Fast R-CNN detector was proposed which solved all the problems mentioned above. It enabled simultaneous training of the detector and bounding box regressor under same network configurations preventing the need to save the data on disk at various stages. It also improved the performance from 58.5% mAP (R-CNN) to 70% while also being almost 200 times faster than R-CNN. However, there was one more bottleneck in this architecture that, if taken care of, could improve the speed further. The bottleneck was the traditional algorithm of selective search that was used to detect the potential proposals in the first stage. Hence, shortly after this, Faster R-CNN was proposed. It was the first end-

to-end deep learning detector. I will expand on this further in a dedicated section as this is the detector that we have used for this work.

One stage detectors

A parallel line of work has been in one-stage approaches to object detection. In 2015, You Only Look Once(YOLO) was proposed as the first one-stage detector. One-stage detectors are designed to keep speed of detection in mind. They are extremely fast. They follow a completely different approach from their two-stage counterparts. While the 2-stage detectors first detect and then verify the proposals, YOLO applies one single neural network to the entire image which divides the image into regions and predicts bounding boxes for each region simultaneously. Even though YOLO showed a lot of improvement in terms of speed, it definitely took a hit on accuracy. It also had difficulty in detecting objects that are too close or too small.

Single Shot MultiBox Detector or SSD was also proposed later that year which solved the problem of detecting variable object sizes and improved the accuracy of 1-stage detectors by introducing the concept of hierarchical features. The idea is that, once the feature extractor extracts the features, all objects whether big or small have dropped in spatial resolution. This works for normal sized objects, however, small objects are too hard to detect in this low resolution features. Hence, SSD uses feature layers at multiple levels to detect all sizes of objects.

Single shot detectors often trade accuracy with real-time processing speed. As real-time speed is less critical for our problem than highly accurate predictions, we do not try such

approaches.

1.4.3 Faster R-CNN architecture

Faster R-CNN[13] is an end to end deep learning 2-stage detector whose variant we have used in this work. I this section I explain how the Faster R-CNN works as proposed by the authors of the original paper. It does not involve the modifications. They will be explained later.

Faster R-CNN consists of two modules. The first module is called Region Proposal Network. As it is clear from the name, this module is a fully convolutional network that finds the potential bounding boxes that might contain the objects called region proposals. The second module is a detector that uses these region proposals and identifies the final regions containing objects. It determines the class of the object and also adjusts the bounding box using a regression network. We further explain how the two models work in the following sections.

Region Proposal Network

Region Proposal Network takes an image as input and outputs object proposals, each with an objectness score. Unlike Fast R-CNN, this is done using a fully convolutional network. RPN consists of a backbone network which takes the image as input and outputs a feature map. Once the features are calculated a small network slides an n x n window over this feature map and for every window it identifies 9 proposals. 9 proposals are a result of

3 types of anchors in 3 different scales. The anchor is a bounding box whose center lies at the center of the window or region. The 3 types of anchors are vertical rectangle, horizontal rectangle and a square. These are then used in 3 different scales to capture different sizes of objects. This is illustrated in figure 1.4.

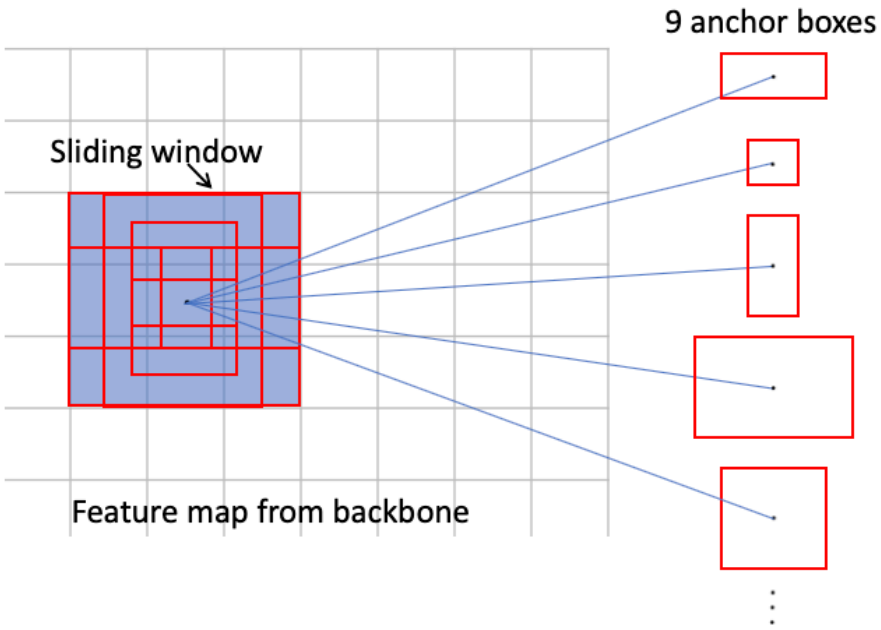

Figure 1.4: Region Proposal network.

The output of this convolution then goes to two small layers viz. classification layer and regression layer that output the objectness(that is whether the bounding box contains an object or not) and the bounding box coordinates respectively. Hence, for every sliding position, we get 2x9 scores from classification layer and 4x9 scores from regression layer. The clasification layer assigns positive class to those anchors that have an Intersection Over Union (IoU) overlap higher than 0.7 with any ground truth box. Once the proposals are generated, they are fed into the Fast R-CNN detector that shares the feature maps gener-

ated by the backbone network. Fast R-CNN takes as input the feature maps from the backbone and a set of object proposals from the RPN. It first applies an ROI pooling layer to the feature maps which extracts fixed length feature vectors for each object proposal. Each feature vector is then fed to fully connected layers that branch into two sibling layers: one produces the clas probability and the other layer produces the bounding box coordinates. The network is shown in the figure 1.5

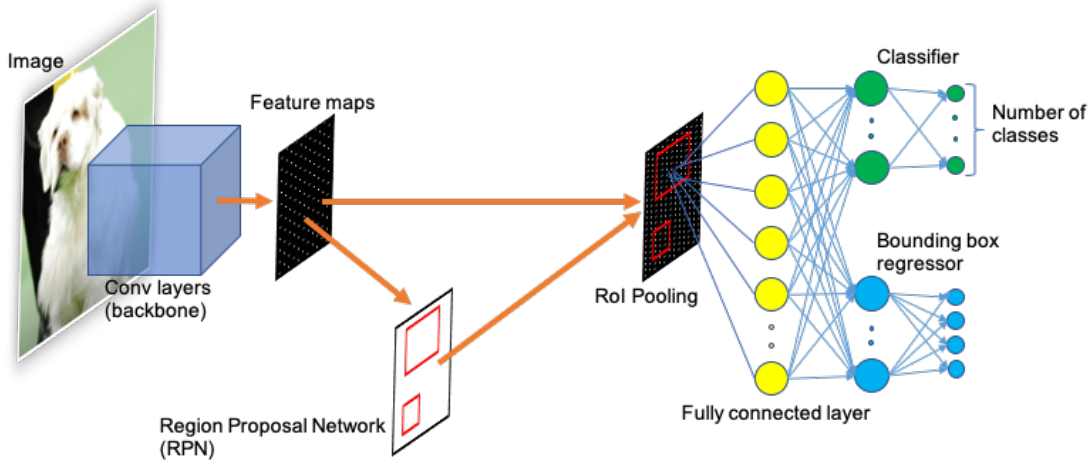

Figure 1.5: Faster R-CNN architecture.

1.5 Attention Mechanism

Attention Mechanism in Machine Learning is a group of techniques that help put the focus on the important part of the input. In machine learning, attention can be of two types: 1. Trainable Attention, which help the model in training to focus on the important things more effectively, and 2. Post-hoc attention, which help humans understand what part of the input contributed towards a decision taken by a pre-trained model. In this section I will

explain trainable attention mechanism for convolutional neural network (CNN) architectures as introduced in "Learn To Pay Attention"[7] by Saumya Jetley et. al.

Let l_i denote the vector of output activations of a convolutional layer at spatial location i and $L = \{l_1, l_2, .., l_i, .., l_n\}$ for $i \in [1, 2, .., n]$ denote the set of feature vectors extracted from a convolutional layer. Then, attention weight for a spatial location which denotes the importance of that location is calculated as:

$$a_i = \frac{exp(c_i)}{\sum_{j=0}^{j=n} exp(c_j)}, i \in \{1...n\}$$

where,

$$c_i = \langle u \, , l_i \rangle$$

is called the compatibility score and has a higher value when the local feature at spatial location i contains the part of the input to be paid attention to and u is a learnable vector. Finally, the output after attention is calculated as:

$$g_a = \sum_{i=1}^{n} a_i . l_i$$

1.6 Related Work

This section covers the work done in breast cancer classification and detection. It briefly expands on various approaches that have been experimented with using the dataset used in this work and also mentions the other work using different datsets.

1.6.1 Simple Classification

Breast cancer screening exam classification is often done through direct classification, rather than with intermediary localization objectives and supervision. Indeed, multi-view convolutional neural networks have been used to directly predict the screening result. For instance,[2] both leverage multi-view methods to predict image-level and exam-level labels with high accuracy. Our results are most directly comparable to these works, as we apply our method to the same dataset. In the Results section, we show that our method that uses more fine-grained supervision compares favorably to this line of work.

1.6.2 Classification by detection

More closely related approach also uses localization information as training data for classification algorithm. In particular,[11] and[14], the two best performing approaches in the DREAM Digital Mammography Challenge, use such approaches. Both methods are also inspired by region proposal networks, with some important differences. These methods used InceptionV3[19] and VGG-16[18] as their backbones architectures, while we use ResNets[6] and their ResNeXt variant[23]. These backbones achieve better accuracies at lower

memory footprints, which is critical to deal with the large resolution of medical images. In addition, unlike us[11] uses deformable conv-nets. Another major difference lies in the scale and higher resolution of the data that we used. We do not directly compare to these methods as the datasets that they evaluate are not public.

1.6.3 Weakly supervised Classification by detection

Another interesting line of work does not use localisation information, but models the tumor localization problem in classifying images. For instance,[16] leverages saliency maps and local patches explicitly, though their method is trained only with image-level labels. Although our method requires labeled data, it performs better.

A more recent work[10] comprises of three models namely Lesion model, Breast model and Case model. Our model closely resembles their Lesion model. It also consists of a detector and an accumulator. However, we use a FasterRCNN as the detector, whereas they use a RetinaNet. We also use an Attention model to accumulate the results from various patches whereas they use a noisy OR. We believe that using a hierarchical network like FPN in the FasterRCNN backbone will suffice for the global context of the image and therefore we do not need breast level or case level models.

By augmenting human performance, AI has the potential to markedly improve productivity, efficiency, workflow, accuracy and speed, both for [physicians] and for patients.

Eric Topol

2

Experiments and Analysis

This chapter describes the data, methodology, experiments, results and analysis carried out for this work. It also expands on the base concepts covered in the previous chapter. All the modifications in the original Faster RCNN architecture are also discussed along with a reasonable explanation.

2.1 Data

Our dataset[22] comprises 229,426 screening mammography exams (1,001,093 images). Each exam contains at least four images, corresponding to the four standard views used in screening mammography: R-CC (right craniocaudal), L-CC (left craniocaudal), R-MLO (right mediolateral oblique) and L-MLO (left mediolateral oblique). with each exam containing the four standard views: L-CC, L-MLO, R-CC, R-MLO, from 141,473 unique patients.

Among these, 5,832 exams had at least one biopsy performed within 120 days of a screening mammogram. Within this set of exams, 985 breasts had malignant findings, 5,556 had benign findings and 234 had both. For these exams, radiologists retrospectively annotated the locations of the biopsied lesions at a pixel level. Figure 2.2 gives a clear picture of these numbers.

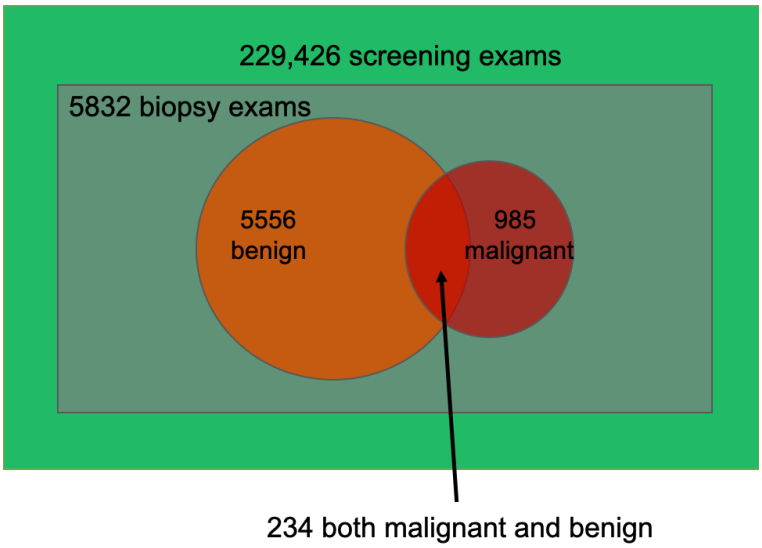

Figure 2.1: Data Venn Diagram

The patients were sorted based on the date of their latest exam and divided into 3 disjoint sets. The oldest 80% form the training set, the next 10% forms the validation set and the latest 10% forms the test set. This leaves us with 186,816, 28,462 and 14,148 exams in the training, validation and test sets respectively.

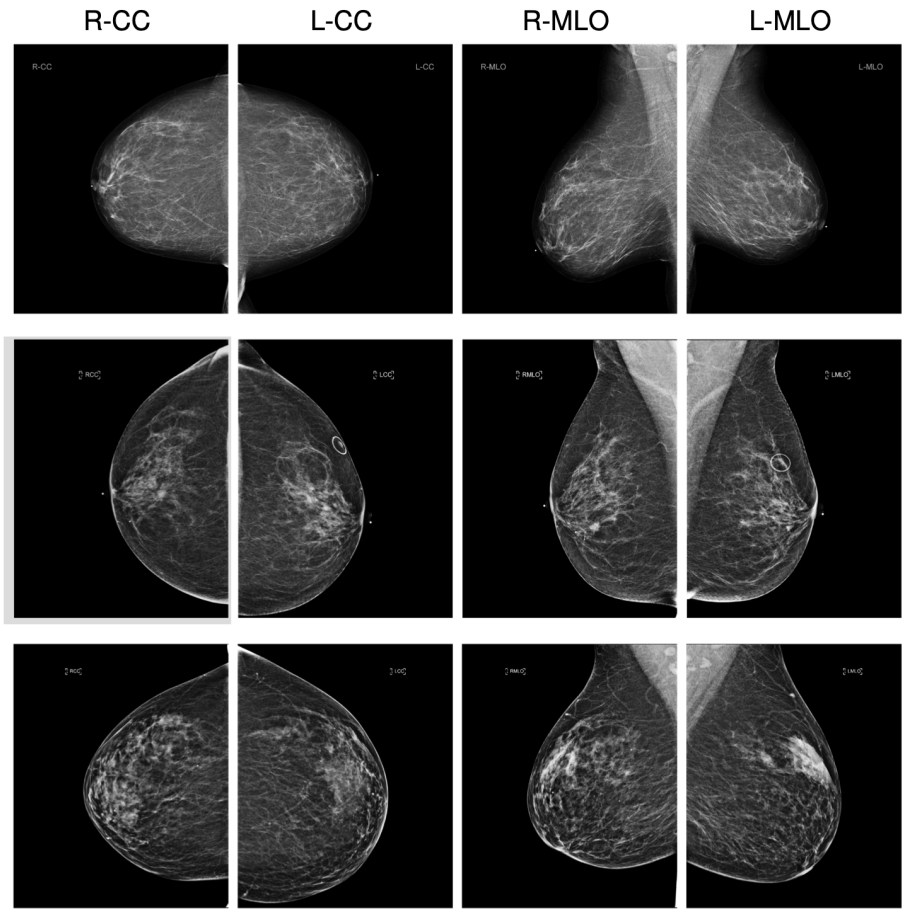

Figure 2.2: Examples of breast cancer screening exams. First row: both breasts without any findings; second row: left breast with no findings and right breast with a malignant finding; third row: left breast with a benign finding and right breast with no findings.[21]

2.2 Methodology

2.2.1 Task Definition

Our task is to create a model that can automatically assign benign and malignant probabilities to the mammogram image. For each mammography exam, we have four labels. For both the left and right breast, we have an annotation for the presence of malignant findings and an annotation for the presence of benign findings. We seek to train an object detection network to predict for each breast the probability that it has benign findings and the probability that it has malignant findings. For each breast, we use the CC and MLO view to make our prediction. We leave to future work the use of left breast views to predict right breast probabilities.

2.2.2 Baseline Model

Our main network is based on Faster-RCNN[13]. This network is trained with a localization objective to detect malignant and benign tumours. To obtain a breast-level prediction of malignancy, we aggregate these localization predictions. We experiment with different approaches for this aggregation which we describe in section 2.2.3.

2.2.3 Modifications from Faster R-CNN

Backbone network

Faster RCNN uses fergus backbone. In this work we experiment with multiple backbones to create the feature maps for the input images. these include Resnet 50, Resnet 101 and Resnext 101. Table 2.1 shows the different results obtained when using different backbones.

Backbone	Resolution	**AUC on development set**	
ResNet 50	2200×3000	0.896	$\pm$ 0.004
ResNet 101	1700×2700	0.890	$\pm$ 0.012
ResNeXt 101	1300×2100	**0.907**	$\pm$ 0.007

Table 2.1: AUC on the development set for different backbones models and images resolutions. In general, increasing resolution and model capacity both helps but for our data ResNeXt-101 offered the best trade-off.

Use of FPN

"Feature pyramid networks for object detection"[9] used feature pyramids to improve detection of objects at different scales. Features are pooled from the successive convolutional layers to build a representation of every region which allows the detector to capture objects of every scale. Figure 2.3, shows the architecture used for the feature pyramidal network. We adopt this change throughout our work as early experiments showed that it significantly improved our results.

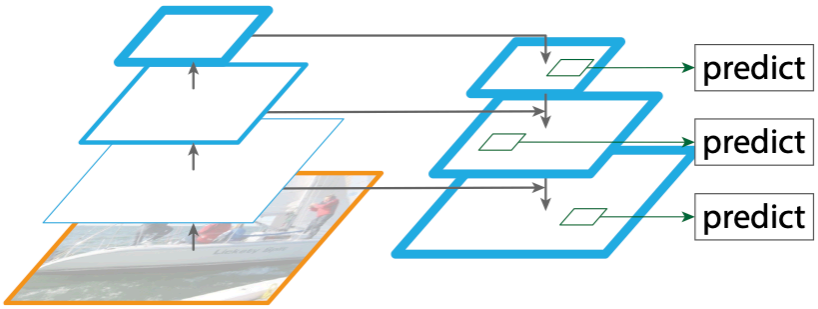

Figure 2.3: Feature Pyramid Network as shown in Lin et al.[9]

RoI Align instead of RoI Pooling

Faster RCNN uses RoI Pooling layer to extract fixed length feature vectors for each project proposal to be fed to fully connected layers of R-CNN. However, we use the RoI Align layer proposed in mask-rcnn[4] rather than the traditional RoI Pooling used in Faster-RCNN. RoI Align fixes the loss of information caused due to quantization of coordinates on feature maps during pooling. Figure 2.4 explains this in more detail. Indeed, mask-rcnn[4] found that RoI Align performed better, even when not doing image segmentation.

IoU Relaxation

We also relaxed the Intersection over Union (IoU) thresholds in our model to account for the difference between natural images and mammograms. Specifically, we relaxed the IoU for foreground objects in the RPN from 0.7 to 0.5, as in[15,12]. Compared to natural images, our data set has (i) noisier annotations with less precise boundaries, and (ii) fewer annotations per image. We also decrease the IoU threshold of the final non-maximum suppression

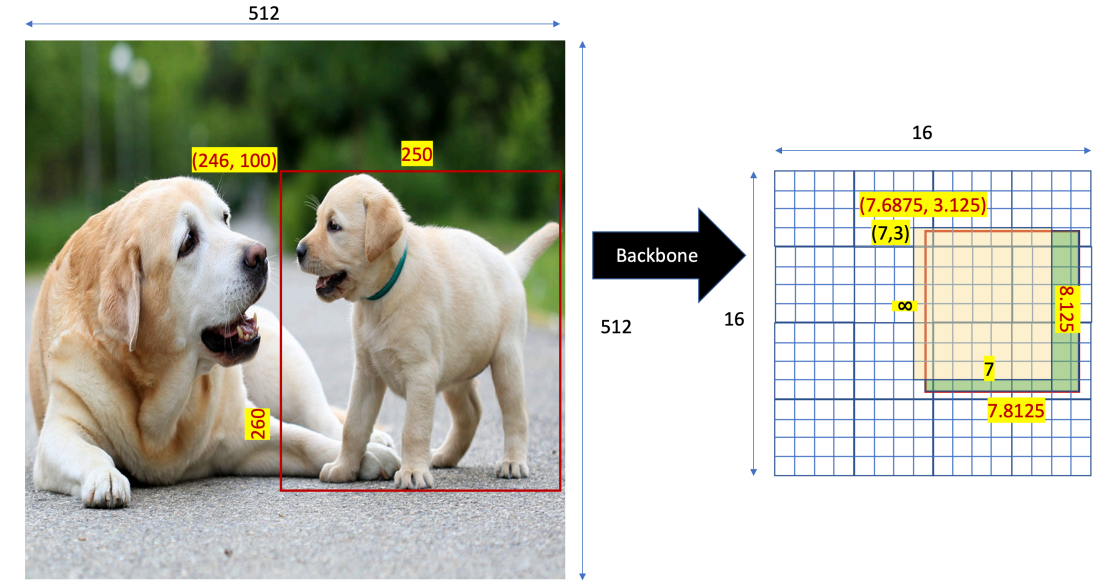

Figure 2.4: When the coordinates of an RoI are mapped to the feature maps, the non integer values are round down to integers causing a loss of data. This is done by RoI Pooling in Faster R-CNN. In this figure, the red bounding box is the actual RoI but the Yellow region is the region that is considered for further pooling. The green region is lost information. Some extra non relevant information can also get added.

to 0.1, following the rationale of[15] that for mammograms, "overlapping detections are expected to happen less often than in usual object detection".

Aggregation Network

In order to get image level labels, we aggregate the predictions from the final Faster R-CNN classifier. We tried a number of ways to aggregate these predictions. One way is to first train the Faster R-CNN network just as an object detector and combine the predictions from different bounding boxes during inference.[1] uses this approach. It combines predictions of various patches in one image to get image level prediction using a max operator and takes a

mean over the two different views to get the breast level prediction. Figure 2.5 explains this in more detail. In this work, we train an attention network on top of regions of interest to get a final image level prediction. We show that incorporating image level information in this way boosts the performance further.

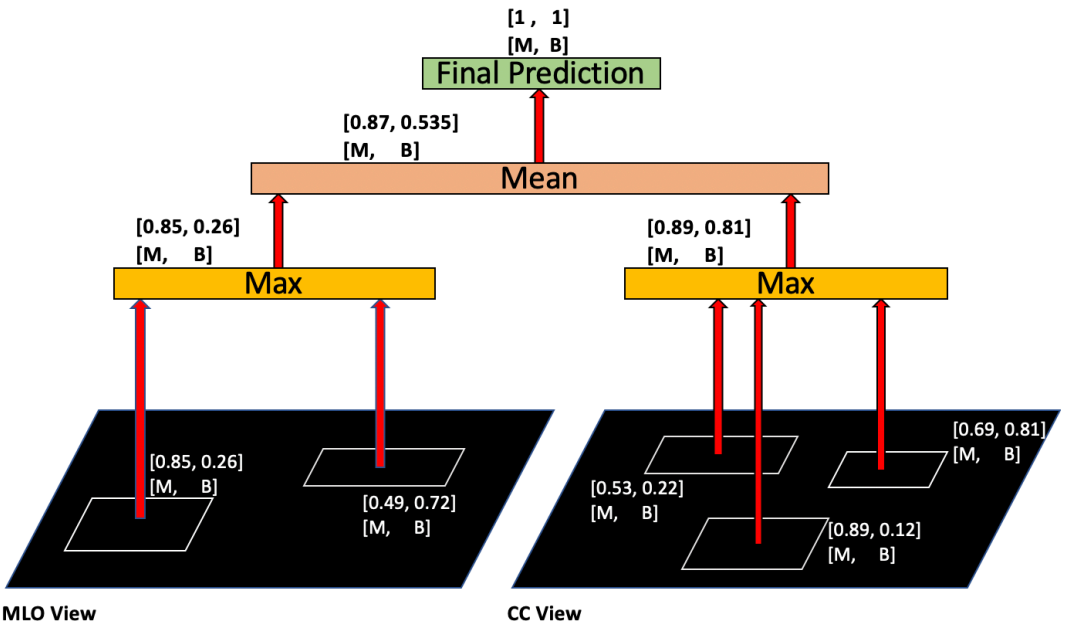

Figure 2.5: This figure illustrates the prediction aggregation as proposed in "Improving localization-based approaches for breast cancer screening exam classification"[1]. M represents probability of presence of malignant finding and B represents benign.

2.3 Experiments

This section discusses all the additional experiments done to build up on the work "Improving localization-based approaches for breast cancer screening exam classification"[1].

2.3.1 Adding global context

"Globally-Aware Multiple Instance Classifier for Breast Cancer Screening" established the importance of global information while classifying a particular patch of medical images. Inspired by that finding, we added global context to our modified Faster R-CNN network during the classification of the proposed regions of interest (RoIs or proposals). Figure 2.6 explains the details of this architecture in more detail. Even though global context was deemed helpful in earlier works, we did not see a gain in our performance by adding global context. One reason for that could be the fact that we are already using pyramidal features that not only capture the objects in different sizes but also provide a global context to detection making adding global context in later stages redundant.

2.3.2 Attention over predictions

To aggregate the information from various patches, we experiment with a small aggregation network that is trained on top of the predictions from the modified faster R-CNN. The entire network is trained in two stages. We first train the Faster R-CNN and then freeze it and train the aggregator network. This network is inspired from the attention network used in "Globally-Aware Multiple Instance Classifier for Breast Cancer Screening"[16]. Let $\hat{\mathbf{y}}_k \in \mathbb{R}^2$ denote the predictions for a patch $\mathbf{x}_k$ from the Faster R-CNN classifier. An attention score α_k is computed for each patch using:

$$\alpha_k = \frac{\exp\{\mathbf{w}^\intercal(\tanh(\mathbf{V}\hat{\mathbf{y}}_k^\intercal) \odot \text{sigm}(\mathbf{U}\hat{\mathbf{y}}_k^\intercal))\}}{\sum_{j=1}^{K}\exp\{\mathbf{w}^\intercal(\tanh(\mathbf{V}\hat{\mathbf{y}}_j^\intercal) \odot \text{sigm}(\mathbf{U}\hat{\mathbf{y}}_j^\intercal))\}}, \tag{2.1}$$

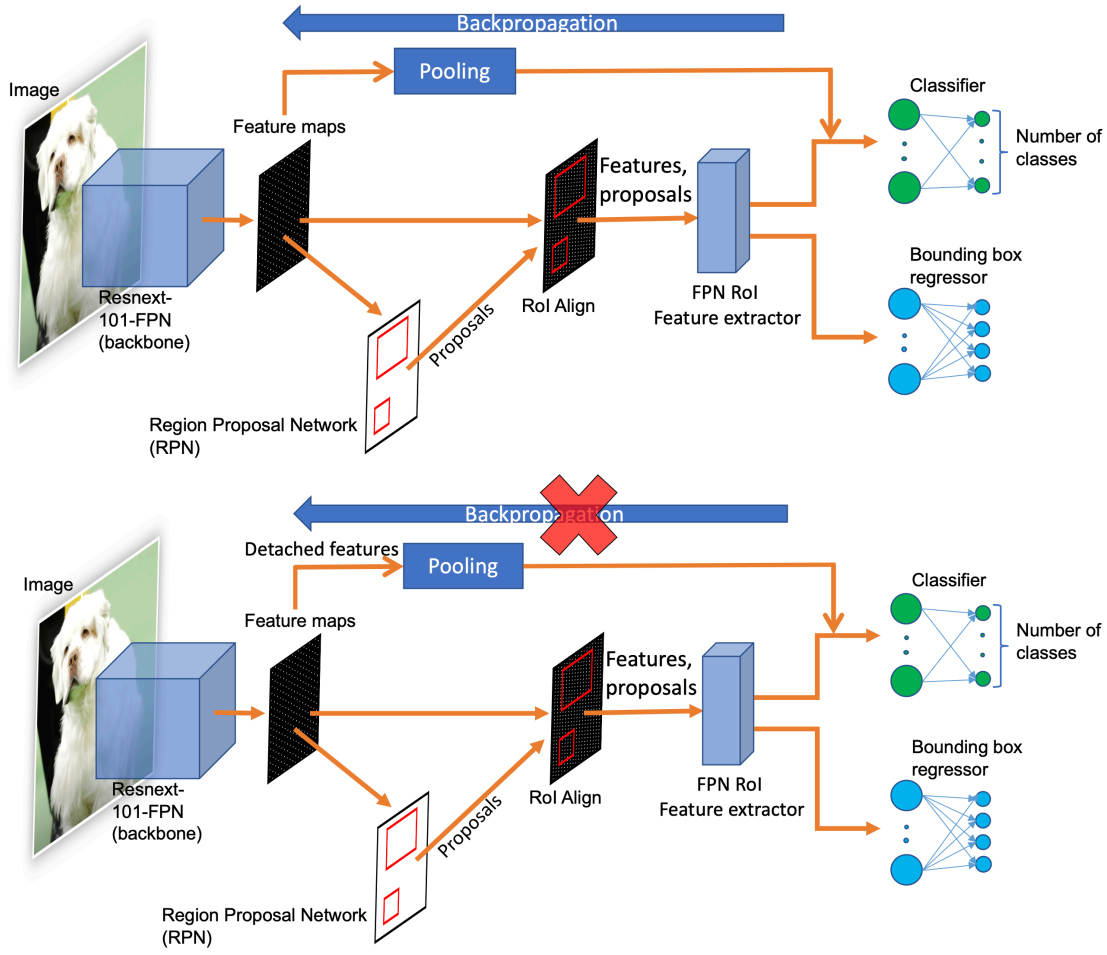

Figure 2.6: Revised architecture of R-CNN with global context. The features are pooled from the backbone output and concatenated with the features from the region of interest. The first experiment lets the loss back-propagate through this new pooling layer and the second one detaches the features from the original feature map before the pooling so the loss does not back-propagate through that branch.

where $\odot$ denotes an element-wise multiplication and $\mathbf{w} \in \mathbb{R}^2$, $\mathbf{V} \in \mathbb{R}^{2\times 2}$, $\mathbf{U} \in \mathbb{R}^{2\times 2}$ are learnable parameters. This process yields an attention-weighted final prediction:

$$\hat{\mathbf{y}}_{image} = \sum_{k=1}^{K} \alpha_k \hat{\mathbf{y}}_k, \tag{2.2}$$

where the attention score $\alpha_k \in [0, 1]$ indicates the relevance of each patch $\mathbf{x}_k$. The results are discussed in the next section.

2.3.3 Attention over feature maps

To aggregate information from various patches, we also trained the attention network on top of the hidden representations of the patches. We train the entire network in a similar fashion where the Faster R-CNN is first trained and then the attention network is trained in second stage. An attention score α_k is computed for each patch $\mathbf{x}_k$ using:

$$\alpha_k = \frac{\exp\{\mathbf{w}^\intercal(\tanh(\mathbf{V}\hat{\mathbf{h}}_k^\intercal) \odot \text{sigm}(\mathbf{U}\hat{\mathbf{h}}_k^\intercal))\}}{\sum_{j=1}^{K} \exp\{\mathbf{w}^\intercal(\tanh(\mathbf{V}\hat{\mathbf{h}}_j^\intercal) \odot \text{sigm}(\mathbf{U}\hat{\mathbf{h}}_j^\intercal))\}}, \tag{2.3}$$

where $\odot$ denotes an element-wise multiplication and $\mathbf{w} \in \mathbb{R}^L$, $\mathbf{V} \in \mathbb{R}^{L\times M}$, $\mathbf{U} \in \mathbb{R}^{L\times M}$ are learnable parameters. Here, we have $L = 1024$ and $M = 64$. This process yields an attention-weighted final prediction:

$$\hat{\mathbf{h}}_{image} = \sum_{k=1}^{K} \alpha_k \hat{\mathbf{h}}_k, \tag{2.4}$$

where the attention score $\alpha_k \in [0, 1]$ indicates the relevance of each patch $\mathbf{x}_k$. The representation $\mathbf{h}_{image}$ is then passed to a fully connected layer with sigmoid activation to generate a final image prediction.

2.4 Results and Analysis

This section discusses the results from all the experiments mentioned in the previous section. We evaluate the model on the task of screening mammography interpretation: predicting the presence or absence of benign and malignant findings in a breast. As each breast is associated with two images (CC and MLO views) and our model generates a prediction for each image, we define breast-level predictions as the mean of the two image-level predictions. For classification performance, we report area under the ROC curve (AUC) on the breast-level and the results are compared with a previous image level network dedicated to mammography (Wu et al., 2019b) and GMIC [16] that evaluate their performance on the same dataset that has been used for this work. Table 2.2 discusses all the results.

Model	AUC (val set)	AUC (test set)
Wu et al. [21] **single model**	-	0.886
Wu et al. [21] **ensemble**	-	0.895
Shen et al. [17] **single model**	-	0.915
Shen et al. [17] **ensemble**	-	0.930
Févry et al. [1] **single model**	91.4	0.908
Févry et al. [1] **ensemble**	-	0.919
Févry et al. [1] **single model with global context**	0.890	-
Févry et al. [1] **single model with detached global context**	0.871	-
Févry et al. [1] **single model with Attention over final predictions**	0.923	-
Févry et al. [1] **single model with Attention over feature maps**	0.934	0.9088

Table 2.2: AUC on the development set for different backbones models and images resolutions. In general, increasing resolution and model capacity both helps but for our data ResNeXt-101 offered the best trade-off.

3

Conclusion

Modified Faster R-CNN with the attention network to detect breast cancer provides two benefits: 1. Boost in the overall accuracy of the classifier, 2. Interpretability. As a helper tool for medical professionals, interpretability is a huge gain provided by this model. The attention network enables the radiologist to assess the predictions of a model well which

allows her to be more confident in its evaluation. Figure 3.1 shows the attention weights highlighting the regions contributing to the malignant and benign findings.

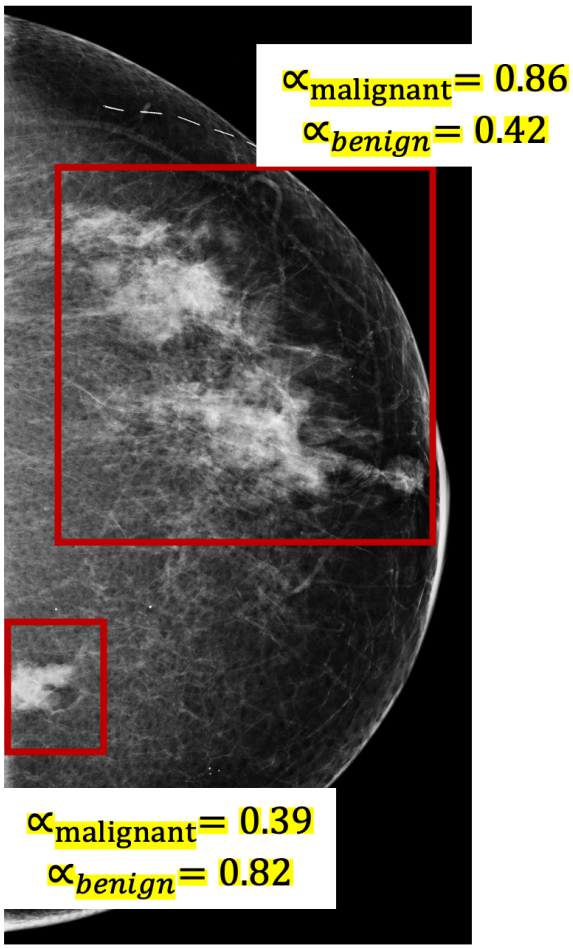

Figure 3.1: The figure display two bounding boxes that had the best attention weights for malignant and benign predictions.

3.1 Limitations

Some limitations of this work are:

- The model utilises an old code-base that makes it harder to make changes for experimentation. It is based on a specific version of PyTorch which is not compatible with the IBM power architecture being used in our research lab. The code-base should be ported to newer version of PyTorch to allow more efficient experimentation.

- The model utilises twice the supervision, i.e. it uses the localised annotations and also the image level labels.

- The model is currently trained in two stages, first the Faster R-CNN is trained and then the attention network is trained. This has two limitations: 1. The model training is slow, and 2. We are not able to fine tune the Faster R-CNN weights on the image level labels. Training the model end to end, might boost performance.

- Our validation metrics varied significantly between check-points. We believe that this is due to (i) the interaction of the components of the training loss, (ii) components of the training loss being only loosely related to the final metric as the model was not trained end-to-end, (iii) a small batch size(1 image per GPU) rendering optimization unstable.

3.2 Future Work

Some future research directions in this area may be:

- Using better and faster backbones for feature generation

- Training the attention model with the Faster R-CNN in an end-to-end manner

- Using object detectors like EfficientDet[20] that have been proposed recently. EfficientDet is much faster and less compute-heavy. It can also be scaled uniformly(in width and depth) as per the resolution of the input images which might be able to give us better results.

- We can also modify the architecture to classify the images without the use of annotations. As annotations are only utilised in the final step to fix the boundaries of the bounding boxes, we can get image level predictions without using the pixel level annotations if we can train the network with attention end to end.

References

[1] Févry, T., Phang, J., Wu, N., Kim, S., Moy, L., Cho, K., & Geras, K. J. (2019). Improving localization-based approaches for breast cancer screening exam classification. *arXiv preprint arXiv:1908.00615*.

[2] Geras, K. J., Wolfson, S., Shen, Y., Wu, N., Kim, S., Kim, E., Heacock, L., Parikh, U., Moy, L., & Cho, K. (2017). High-resolution breast cancer screening with multi-view deep convolutional neural networks. *arXiv preprint arXiv:1703.07047*.

[3] Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 580–587).

[4] He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 2961–2969).

[5] He, K., Zhang, X., Ren, S., & Sun, J. (2015). Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine intelligence*, 37(9), 1904–1916.

[6] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *CVPR*.

[7] Jetley, S., Lord, N. A., Lee, N., & Torr, P. H. (2018). Learn to pay attention. *arXiv preprint arXiv:1804.02391*.

[8] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436–444.

[9] Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2117–2125).

[10] McKinney, S. M., Sieniek, M., Godbole, V., Godwin, J., Antropova, N., Ashrafian, H., Back, T., Chesus, M., Corrado, G. C., Darzi, A., et al. (2020). International evaluation of an ai system for breast cancer screening. *Nature*, 577(7788), 89–94.

[11] Morrell, S., Wojna, Z., Khoo, C. S., Ourselin, S., & Iglesias, J. E. (2018a). Large-scale mammography cad with deformable conv-nets. In *Image Analysis for Moving Organ, Breast, and Thoracic Images* (pp. 64–72). Springer.

[12] Morrell, S., Wojna, Z., Khoo, C. S., Ourselin, S., & Iglesias, J. E. (2018b). Large-scale mammography cad with deformable conv-nets. In *Image Analysis for Moving Organ, Breast, and Thoracic Images*.

[13] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems* (pp. 91–99).

[14] Ribli, D., Horváth, A., Unger, Z., Pollner, P., & Csabai, I. (2018a). Detecting and classifying lesions in mammograms with deep learning. *Scientific reports*, 8(1), 4165.

[15] Ribli, D., Horváth, A., Unger, Z., Pollner, P., & Csabai, I. (2018b). Detecting and classifying lesions in mammograms with deep learning. *Scientific Reports*.

[16] Shen, Y., Wu, N., Phang, J., Park, J., Kim, G., Moy, L., Cho, K., & Geras, K. J. (2019). Globally-aware multiple instance classifier for breast cancer screening. In *International Workshop on Machine Learning in Medical Imaging* (pp. 18–26).: Springer.

[17] Shen, Y., Wu, N., Phang, J., Park, J., Liu, K., Tyagi, S., Heacock, L., Kim, S., Moy, L., Cho, K., et al. (2020). An interpretable classifier for high-resolution breast cancer screening images utilizing weakly supervised localization. *arXiv preprint arXiv:2002.07613*.

[18] Simonyan, K. & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[19] Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2818–2826).

[20] Tan, M., Pang, R., & Le, Q. V. (2019). Efficientdet: Scalable and efficient object detection. *arXiv preprint arXiv:1911.09070*.

[21] Wu, N., Phang, J., Park, J., Shen, Y., Huang, Z., Zorin, M., S. Jastrzębski, T. Févry, Katsnelson, J., Kim, E., Wolfson, S., Parikh, U., Gaddam, S., Lin, L. L. Y., Ho, K., Weinstein, J. D., Reig, B., Gao, Y., Toth, H., Pysarenko, K., Lewin, A., Lee, J., Airola, K., Mema, E., Chung, S., Hwang, E., Samreen, N., Kim, S. G., Heacock, L., Moy, L., Cho, K., & Geras, K. J. (2019a). Deep neural networks improve radiologists' performance in breast cancer screening. *arXiv:1903.08297*.

[22] Wu, N., Phang, J., Park, J., Shen, Y., Kim, S. G., Heacock, L., Moy, L., Cho, K., & Geras, K. J. (2019b). *The NYU Breast Cancer Screening Dataset v1.0*. Technical report. Available at https://cs.nyu.edu/~kgeras/reports/datav1.0.pdf.

[23] Xie, S., Girshick, R., Dollár, P., Tu, Z., & He, K. (2017). Aggregated residual transformations for deep neural networks. In *CVPR*.

[24] Zou, Z., Shi, Z., Guo, Y., & Ye, J. (2019). Object detection in 20 years: A survey. *arXiv preprint arXiv:1905.05055*.