Using Deep Learning for Histopathological Cancer Detection

Assion Kuyona Tetteh
University of Maryland
College Park - MD
atetteh3@umd.edu

Hemanth Joseph Raj
University of Maryland
College Park - MD
hemanth1@umd.edu

Noah Isbell
University of Maryland
College Park - MD
nisbell@umd.edu

Abstract

Deep Learning is in the forefront of a huge number of applications these days. These applications have a wide range of use from healthcare computer vision, financial analysis, fraud detection, etc. Within the healthcare domain Deep Learning has a lot of applications like drug discovery, patient monitoring, medical imaging, and diagnostics. For this project, we are choosing the field of medical imaging, specifically breast cancer detection. We are developing a custom CNN architecture to detect and classify normal and cancerous breast tissue. We are using a publicly available dataset comprised of the histopathological slide images of regular and cancerous breast tissue. After training our model on this data, we will be using it for Breast Cancer Detection. The model we finalized converged well with a good Area under the Curve even with augmented data.

I. Introduction

Artificial Intelligence is used in a variety of fields, such as housing, finance, healthcare, and education. It is fascinating that a simple model based on linear equations can be used for complex tasks like classification. In particular, the use of deep learning, a type of AI that uses artificial neural networks, has become more widespread in recent years.

AI plays a crucial role in perception tasks, such as image classification. A straightforward neural network can accurately classify images into different categories, even if the images contain hidden, occluded, or unusual features. This is because the network is able to learn from the data, rather than relying on manually defined filters for feature detection. In contrast, traditional computer vision algorithms require these filters to be defined for each class of interest, which is difficult or impossible to do for all possible features and variations. As a result, neural networks can outperform traditional algorithms in many cases.

If we put together the best features of conventional Computer Vision algorithm, i.e., the different filters which work towards detect different features of an image with the computational strength of neural networks, we get an effective algorithm for image classification, Convolutional Neural Networks. CNNs use layers of filters of different kernel sizes to help detect certain features which are trained into the network. This method to classify data, especially images help to improve the efficiency of our predictions. This versatility is why are using CNNs for our application as we need to perform binary classification of images and since the application is cancer detection, we need it to be highly accurate because it can lead to a great deal of ethical and legal problems if we don't.

Cancer Detection is one of the areas of healthcare where Deep Learning can play a huge role. It is the leading cause of deaths in the world and one in six deaths can be attributed to cancer [1]. As the world leading cause of death and with over twelve different types, it has become very important to understand more the more behavioral pattern of cancer cells in human body and how to detect them early for possible cure if not prevent them. One defining feature of cancer is the rapid creation of abnormal cells that grow beyond their usual boundaries, which can then invade adjoining parts of the body and spread to other organs known as metastases are the primary cause of death from cancer. Our focus will be on Breast Cancer which is one of the most common forms of cancer. For a long time now, oncologists depend on the assessment of scans made by pathologists. We propose to make use of AI to help aid the process. The increasing adoption of deep learning across healthcare domains together with the availability of highly characterized cancer datasets has accelerated research into the utility of deep learning in the analysis of the complex biology of cancer. Our main objective for this project is to apply the knowledge and various technique of deep learning to detect a cancer cell among other human cells.

The models trained on non-augmented images converge to a solution at a much faster rate.

An approach that was considered but not tested, was to train a model to a high degree of accuracy. Then use a transfer learning approach as the model already can detect cancer in the center of a slide image. The weights could be locked, and additional layers could be added to detect cancerous cells in the augmented images.

II. Literature Review

Experimentation with using Machine Learning and Deep Learning techniques for Cancer Detection have been taking place for a long time now. One of the earliest experimentations of using neural networks for Breast Cancer detection presents a new method for analyzing texture in microscope images of biopsy samples [3]. The technique uses fuzzy co-occurrence matrices and a feature extraction algorithm to classify the images into three risk groups. In [4], a paper written in 2007, potential benefits of using support vector machines (SVMs) and radial basis function (RBF) for detecting breast cancer are explored. 1-norm and 2-norm C-SVM (L1-SVM and L2-SVM) is used to develop a grid search method based on gradient descent and validation error estimate (GDVEE) to improve detection accuracy. Other Machine Learning based solutions have been used. In this paper [5], the authors describe a method for distinguishing between benign and malignant breast tumors using a combination of support vector machines, K-nearest neighbors, and probabilistic neural networks classifiers. They use signal-to-noise ratio feature ranking, sequential forward selection-based feature selection, and principal component analysis for feature extraction.

Many Deep Learning approaches have been explored too. In this paper [6], the authors introduce a new deep learning network based on the ResNet architecture that is designed to identify metastatic cancer from cancer scan images. To make their model more reliable and accurate, they also apply Test Time Augmentation. The results of their experiments show that the proposed ResNet-based model outperforms previous VGG16 and VGG19 models, achieving state-of-the-art performance on this task.

The current state-of-the-art in terms of cancer detection on the Histopathological Cancer Detection dataset from Kaggle (modified PatchCamelyon dataset by removing duplicates) is in the area of dense steerable filters for CNNs. The paper [7] presents a novel approach to incorporating rotational symmetry into Convolutional Neural Networks (CNNs) for histology image analysis. The proposed method, called Dense Steerable Filter CNNs (DSF-CNNs), uses group convolutions with multiple rotated copies of each filter in a densely connected framework. By allowing the CNN to be rotation-equivariant, the need to learn this set of transformations from the data is eliminated, freeing up model capacity and reducing the risk of overfitting. Additionally, the use of steerable basis filters enables exact rotation and reduces the number of trainable parameters compared to standard filters. The authors provide the first in-depth comparison of different rotation-equivariant CNNs for histology image analysis and demonstrate the superiority of DSF-CNNs. They show that this approach achieves state-of-the-art performance on three different tasks in computational pathology, using significantly fewer parameters than existing methods.

III. Problem Statement

In the current widely practiced protocol, an oncologist looks at medical imaging, either Contrast CTs, MRIs or PET scans and use their medical acumen and judgment to tell with a certain degree of confidence if a person has cancer or not. Provided that an oncologist must be quite skilled in their craft of looking at films and making decisions, there still is room for probable error of a misdiagnosis. This error can have a huge implication on not only the mental health of a patient but also it will affect the credibility of the oncologist. In some cases, there may be financial repercussions too.

Our proposition here is to build a Deep Learning model based on Convolutional Neural Networks to help create a model which can classify histopathological slide images of lymph node sections and determine if a certain image has metastatic tissues. This solution is not intended to replace the medical practitioner in making cancer diagnosis, but to aid and supplement their diagnosis.

The kind of cancer in focus here is Breast Cancer, one of the most common forms of cancer.

III.1. Input/output of the model

Deep-learning algorithms such as CNNs have been designed to perform computer vision tasks. In simplified terms, a typical CNN tiles an input image with small matrices known as kernel nodes or filters. Each filter encodes a pixel intensity pattern that it ‘detects’ as it convolves across the input image.

A multitude of filters encoding different pixel intensity patterns convolve across the image to produce two-dimensional activation maps of each filter.

The algorithm takes in images as input from the data and outputs the predicted ID label, 1 or 0, for the images. The image ID and the prediction label is used to check the accuracy of the model

III.2. Evaluation Metric

The data is organized such that the CSV files have the image ID and the label to go along with it. When an image is used in training, the predicted label for that image will be compared with the actual label to evaluate its efficiency. This metric will also be applied on the testing set to evaluate its efficiency as well. Another metric we will be using is the Area under ROC curve. It is a plot of TP vs FP (True Positive vs False positive rate) at different threshold values.

AUC is nothing but the probability that the designed classifier will randomly choose a positive value and a negative value and rank the positive instance higher than the negative instance.

III.3. State-of-the-art vs our approach

Currently, the state-of-the-art approaches for Deep Learning for Cancer detection are all using a method called as the Dense Steerable Filters for CNNs (DSF-CNN). Also used is transfer learning by first training the model of existing models such as xception and VGG-19 and then performing some post processing to achieve the best results.

Our approach is to use topics learned in class like sequential Neural Network API from PyTorch or Tensorflow and apply it on our own architecture and work towards achieving similar results.

IV. Data

IV.1. Dataset Information

In this project we will be using the *Histopathological Cancer Detection* dataset offered on the Kaggle Competion [2] website for the same. It is a slightly updated version of the PatchCamelyon (PCam) dataset which is a vast collection of small image patches taken from larger digital pathology scans of breast lymph node sections.

The images in this dataset are broken down into train and test sets. All images are labeled as either 1 or 0 where 1 refers to the image being that of cancer and 0 to be otherwise. A positive label on an image indicates that the center 32x32 pixel region of a patch contains at least one pixel of tumor tissue. This helps in creating models that do not use any zero-padding when using Convolutional Neural Networks

The breakdown is as follows:

1. Training: 153k (0.9) images
2. Validation: 17k (0.1) images
3. Testing: 57.5k images

IV.2. Exploratory Data Analysis

We investigated the dataset to see if the sampled data in training and validation and testing are homogeneously arranged, i.e., equal distribution of images with label 1 (cancerous) and label 0 (benign) in all sets of data.

The images in the dataset are of 96x96 pixels resolution. The information on the data suggests that we will find cancer tissue cells in the center most 32x32 pixels of each image. So, we will be looking only at that region of the image.

Checking the head of the data and the info, to understand what type of data we are dealing with.

	id	label
0	f38a6374c348f90b587e046aac6079959adf3835	0
1	c18f2d887b7ae4f6742ee445113fa1aef383ed77	1
2	755db6279dae599ebb4d39a9123cce439965282d	0
3	bc3f0c64fb968ff4a8bd33af6971ecae77c75e08	0
4	068aba587a4950175d04c680d38943fd488d6a9d	0

Fig. 1 Data head

Exploring further we can see about further statistics in the data. One thing to note here is the mean of the data is about 0.405031. This means that the negative and positive samples in the data aren't split equally.

	label
count	220025.000000
mean	0.405031
std	0.490899
min	0.000000
25%	0.000000
50%	0.000000
75%	1.000000
max	1.000000

Fig. 2 Data Description

This assumption is confirmed when we plot the data histogram. We see that there are more negative samples than positive samples.

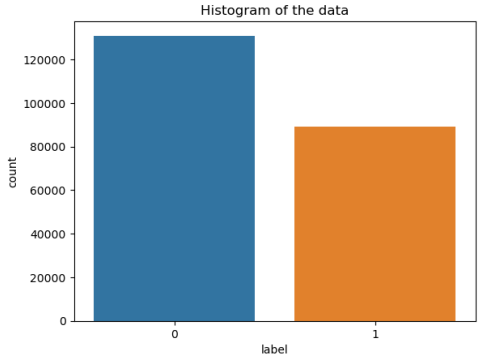

Fig. 3 Data Histogram

Another way to visualize the data is by plotting a pie chart of the data.

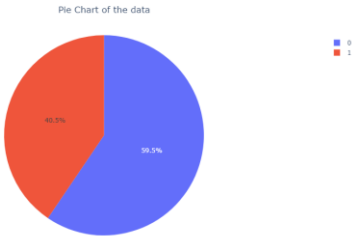

Fig. 4 Data Pie Chart

Next, let's see how to images themselves look. Here we randomly sample images, so we don't really have a chance to choose which images we want. As we can see, the center 32x32 portion of the pixel is highlighted

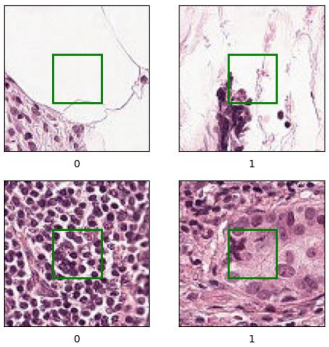

Fig. 5 Random Image sampling

Now we can selectively sample images from the test set of normal and cancerous images. This gives us more control over what images we want to see. Once again, we can see the center 32x32 portion of the image to see if the image has cancer cells or not.

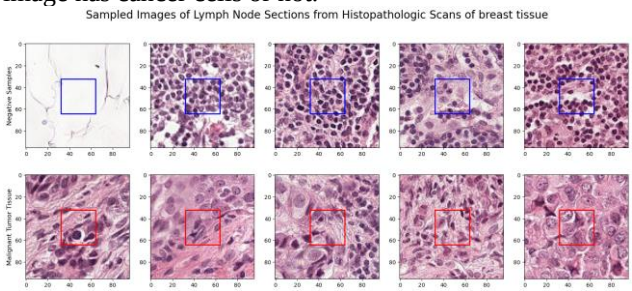

Fig. 6 Negative and Positive Samples of the data

V. Methods

Our goal was to increase the accuracy of detecting malignant cancer cells anywhere in a medical slide image. After spending a significant amount of time looking into the generation of the image dataset, it was determined that the images are part of a Whole-slide-image (WSI) dataset that stacks 40x, 10x and 1x magnification into the image. The original dataset couldn't be obtained, even after requesting access. The dataset that was obtained is a pre-processed version of the images. The images containing cancer cells have them in the center of the slide image.

We believed the best approach was to use augmented slide images. A known accurate CNN model was trained for 10 epochs with the non-augmented images. The trained model was detecting cancer cells with approximately 87% accuracy on non-augmented slide images. Testing the trained model with augmented slide images, the detection accuracy drops to approximately 70% accuracy. The models trained on non-augmented images converge to a solution at a much faster rate.

An approach that was considered but not tested, was to train a model to a high degree of accuracy. Then use a transfer learning approach as the model already can detect cancer in the center of a slide image. The weights could be locked and additional layers could be added to detect cancerous cells in the augmented images.

VI. Experiments

To have a model that will detect cancerous cells when not centered and from noisier data, we first need to generate noisy images. We took advantage of some of the built-in Keras functionality in Tensorflow. The load image function has built in image augmentation capabilities. The images were augmented randomly with random augmentations being applies. The images were being shifted, rotated, flipped, color and contrast shifted. The augmented images were used to test trained CNN models on regular slide images. There was a significant drop in detection accuracy on the 'non-clean' images.

With CNN's filtering technique working very well on images, we wanted to build upon the existing premise. The first few models we selected to test, were taking 15-25 minutes/epoch. The very long training time slowed down our progress. A function was created to made train and evaluate models and their parameters, checking their performance. After numerous modes were created and 3+ days of continuous testing the system crashed. All generated data was lost, and no plots or data were able to be captured. Several of the CNN models created, such as the one below, wouldn't generalize to new data. The model was 5-D [Conv -> ReLU -> MaxPool -> Dropout] then fed into a fully-connected neural network. This model ended up being determined to not generalizing well to new data as a result of very high early dropout rates. The first two CNN layers had 80% & 60% chances of dropping respectively. For that specific model the tested parameters for dropout weren't below 30% until the fifth layer which was still 10% chance.

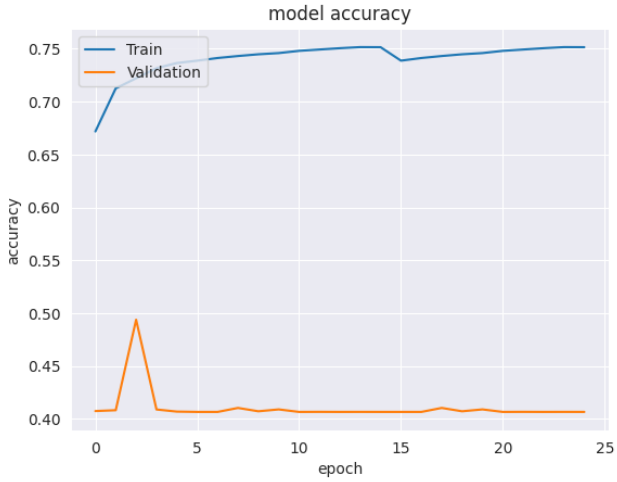

Fig. 7 Model Accuracy

Fig. 8 Model loss

The observed outputs during the lost and not testing was used in the subsequent decisions when creating our 'fourth' model. The CNN network was made to detect many early small features in the image. The model had 256 filters for the first two layers. The model consisted of 4x[Conv -> BatchNorm -> Conv -> ReLU -> BatchNorm -> MaxPool -> Dropout] layers, with 2x2 kernals sizes and filters as follows: 256, 256, 128, 1268, 64, 64, 128, 128. Note: The filter of size 1268 was a typo in the code but the actual value used on accident for the filter.

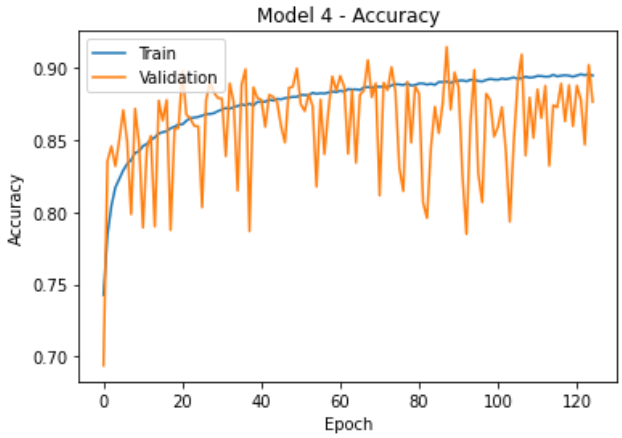

Fig. 9 Model 4 accuracy

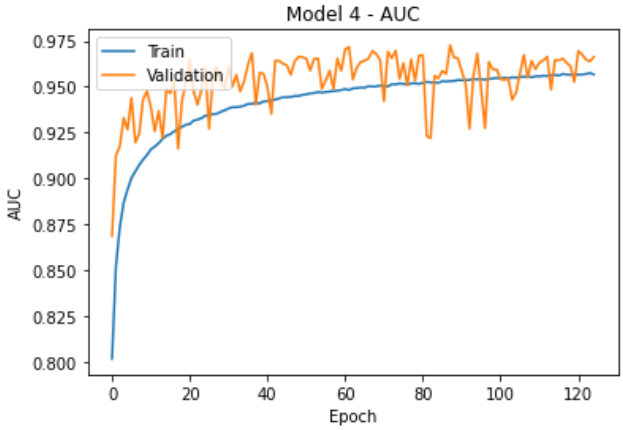

Fig. 10 Model 4 Area Under the ROC Curve

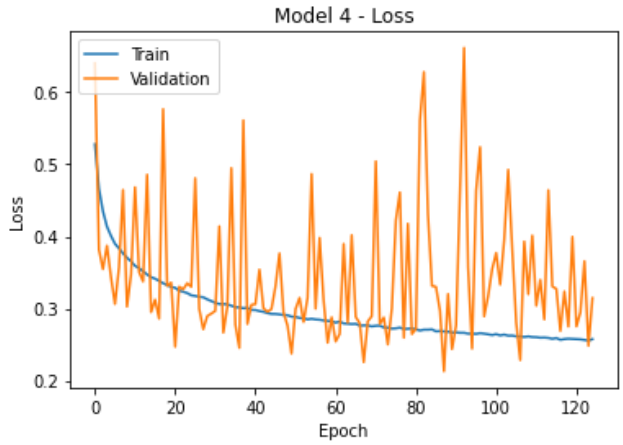

Fig. 11 Model 4 loss

As can been interpreted from the graphs, the model was still learning and starting to converge. If the model had more time to train it would eventually be detecting with a higher accuracy. We were working on other model concurrently to try and reduce the jitter in the convergence of the model and speed up training time. The other models

being investigated put more emphasis on the number of neurons in the deeper layers. The first layer has 8 16x16 filters with up to six CNN layers before 4 fully-connected layers. That model trains at 3 minutes/epoch which is a significant improvement and the jitter during training is reduced.

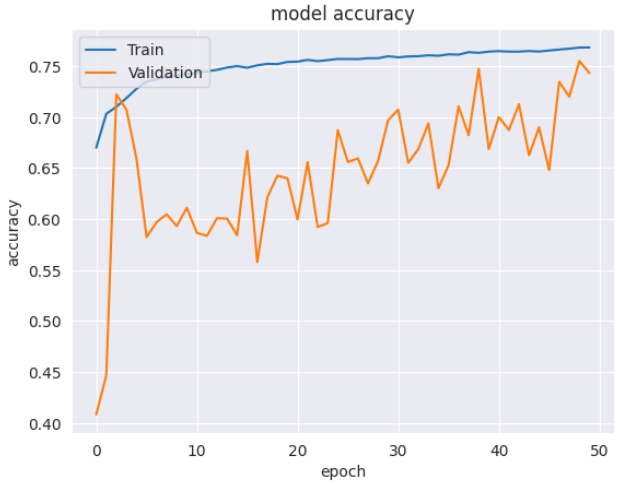

Fig. 12 Model Accuracy

Fig. 13 Model loss

VII. Limitations

A major challenge with implementing deep learning into clinical practice is the 'black box' nature of the models. High-stake medical decisions, such as diagnosis, prognosis, and treatment selection, require trustworthy and explainable decision processes. Most deep learning models have limited interpretability, meaning it is very difficult to dissect a neural network and understand how millions of parameters work simultaneously. Some even argue that more interpretable models such as decision trees should be ultimately preferred for making medical decisions. An alternative approach is explained ability—mathematical quantification of how influential, or 'salient', the features are towards a certain prediction. This information can be used to 'explain' the decision-making process of a neural network model and identify features that contribute to a prediction. This knowledge can enable resolution of potential disagreements between deep learning models and clinicians and thus increase trust in deep learning systems. Moreover, deep learning models do not always have perfect performance due to either imperfect training or systematic errors caused by bias within the models themselves, which can result from the training data not being representative of the data where it is applied. In these circumstances, explain-ability can assist clinicians in evaluating predictions. While some explain ability methods were developed specifically for neural networks others offer a more model- and data-agnostic solution. The Team has used TensorFlow's Image Data Generator for the data augmentation which allows for numerous image augmentation performed randomly. ImageDataGenerator class allows you to randomly rotate images through any degree between 0 and 360 by providing an integer value in the rotation range argument. The image dataset is split into training and validation sets with 85% and 15% respectively. The image dataset is normalized, allowing for random right/left/up/down shifting with the data wrapping around to fill the space. This is necessary because data augmentation makes the model more robust to slight variations, and hence prevents the model from overfitting. It is neither practical nor efficient to store the augmented data in memory, and that is where the ImageDataGenerator class from Keras (also included in the TensorFlow's high level api: tensorflow.keras) comes into play. ImageDataGenerator generates batches of tensor image data with real-time data augmentation. But what about the points where we don't have any value? This is the default option where the closest pixel value is chosen and repeated for all the empty values. This mode instead creates a 'reflection' and fills the empty values in a reverse order of the known values rather. The idea behind using a Keras generator is to get batches of input and corresponding output on the fly during training process by reading images and getting corresponding label vectors and then feeding this set to the GPU for training step, but the problem we faced was memory requirement for the standard Keras generator. The in-memory generator creates copies of the original data as well as converting it from dtype uint8 to float64. On the other hand, the Keras generator to read from directory expects images in each class to be in an independent directory which is not possible in multi-label problems, or segmentation problems. Training with augmented images was taking a significant amount of time. We ran through 10 epochs (3 hours 12 minutes) of training on non-augmented images and 2 epochs of augmented images (~40 minutes). With 1-10 Epochs on CNN there was a

significant accuracy loss (~15%) when training on non-augmented images and testing on augmented images.

VIII. Conclusions

Even though there are few challenges with deep learning implementation for Cancer Detection, we do observe that the advantages and potential technical knowledge is highly encouraging. After changing model to have more channels to allow better feature detection in the earlier CNN layers, both datasets now take about the same amount of time to train. In 10 epochs the model was trained to 91% accuracy with a generalization accuracy of 80 We then decided to push for training a model for 125 epochs using the augmented image data which took about 38 hours.

IX. Results

After training our model on this data, we used it for Breast Cancer Detection. The model we finalized converged well with a good Area under the Curve even with augmented data. The trained model would detect with a 87% accuracy rate for both augmented slide images and non-augmented slide images. After training the model had an AUC on the trained data of 0.9567 and an AUC of 0.9663 on new data. The state-of-the-art [9] currently (DSF-CNN) achieves an AUC around 0.975.

The model even after 125 epochs was still working on converging. With more training the model shows the potential of having a better detection accuracy.

We achieved similar detection accuracy results as the model we based our solution off of [8]. It is worth noting that our model achieved that accuracy for both augmented and non-augment images, whereas the model we referenced [8], when tested on augment data, the detection accuracy was between 65% and 75%. The range is due to the random generation of augmented data, the more the augmented data the worse the model performs.

References

[1] World Health Organization: Cancer Factsheet. https://www.who.int/news-room/fact-sheets/detail/cancer#:~:text=Cancer%20is%20a%20la

[2] Histopathological Cancer Detection Dataset – Kaggle. https://www.kaggle.com/c/histopathologic-cancer-detection

[3] H. D. Cheng, C. H. Chen and R. I. Freimanis, "A neural network for breast cancer detection using fuzzy entropy approach," Proceedings., International Conference on Image Processing, 1995, pp. 141-144 vol.3, doi: 10.1109/ICIP.1995.537600.

[4] Iranpour, Majid, Sanaz Almassi, and Morteza Analoui. "Breast Cancer Detection from FNA Using SVM and RBF Classifier." (2007).

[5] A. Osareh and B. Shadgar, "Machine learning techniques to diagnose breast cancer," 2010 5th International Symposium on Health Informatics and Bioinformatics, 2010, pp. 114-120, doi: 10.1109/HIBIT.2010.5478895.

[6] Z. Zheng, H. Zhang, X. Li, S. Liu and Y. Teng, "ResNet-Based Model for Cancer Detection," 2021 IEEE International Conference on Consumer Electronics and Computer Engineering (ICCECE), 2021, pp. 325-328, doi: 10.1109/ICCECE51280.2021.9342346.

[7] Graham, S., Epstein, D., & Rajpoot, N. (2020). Dense Steerable Filter CNNs for Exploiting Rotational Symmetry in Histology Images. *arXiv*. https://doi.org/10.48550/arXiv.2004.03037

[8] Mattison Hineline, “Cancer CNN Detection”, Kaggle, https://www.kaggle.com/code/mattison/cancer-cnn-detection

[9] Papers with code, “Breast Tumour Classification on PCam”, Leaderboard https://paperswithcode.com/sota/breast-tumour-classification-on-pcam