Breast Cancer Detection from Mammographic Images using Deep Learning

Aswini Pusuluri

Department of Applied Data Science, San Jose State University

DATA 270: Data Analytics Processes

Dr. Eduardo Chan

December 10, 2021

Abstract

Breast cancer is one prominent kind of cancer among different cancers that had a significant share of contribution to the deaths of individuals, especially women. Early identification and diagnosis of breast cancer has become quintessential in addressing the problem. One of the cost effective and efficient screening techniques that has been widely adopted is Mammography. Some of the challenges in diagnosis of breast cancer using mammography screening are incorrect interpretations by radiologists and false negatives during screening. Conventional machine learning models had limitations like feature extraction requiring manual intervention and inoperable on raw images. This research project is aimed at minimizing the above-mentioned shortcomings and enhance the performance and efficiency of breast cancer detection using Deep Learning pipeline. The proposed pipeline is constructed from Convolutional Neural Network (CNN) models with different training approaches performed on the Mammography Image Analysis Society (MIAS) mammogram image dataset. Preprocessing, augmentation, and segmentation are performed on the raw images before providing to three CNN models to perform feature extraction and classification of images into normal, malignant, and benign. Modified CNN model is trained from scratch while AlexNet and Residual Network-50 (ResNet-50) are based on transfer learning. The proposed models are evaluated for performance and accuracy based on confusion matrix and Area Under the Receiver Operating Characteristic Curve (AUC-ROC), which determined ResNet-50 to be more accurate and efficient with 98.6% accuracy, 97% precision, 100% recall and 99% F1-Score.

Keywords: Mammography, breast cancer, deep learning, MIAS, segmentation, augmentation, CNN, AlexNet, ResNet-50

1. Introduction

1.1 Project Background and Execute Summary

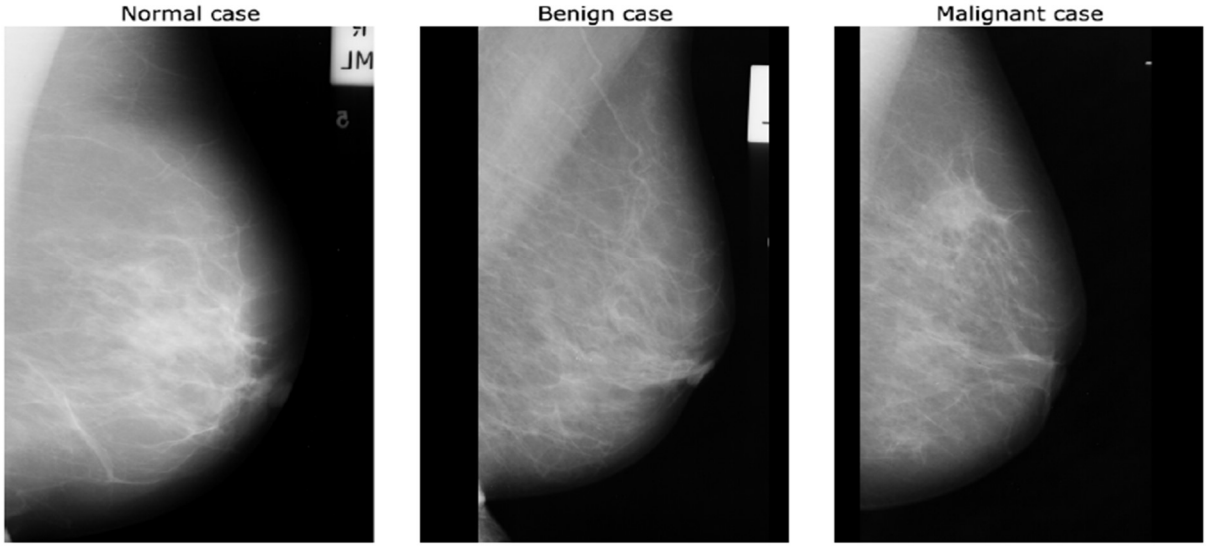
Breast cancer is a renowned form of cancers which is mostly detected among women in America. According to U.S. Breast Cancer Statistics (2021), it is expected that of all the cancers that will be newly diagnosed in women 30% will be breast cancers. The above estimate is translated to almost 281,000 novel invasive breast cancer cases and about 49,200 non-invasive breast cancer cases among American women. Breast cancer is diagnosed primarily by the nature of the tumors, whether they are benign or malignant. It is considered as positive if the tumors are malignant and negative if the tumors are benign in nature. There are different kinds of detection techniques that are available today in the diagnosis of breast cancer like Mammography, Breast Ultrasound scan, Magnetic resonance imaging (MRI), Photoacoustic Imaging, Computed Tomography (CT) Scan, Nuclear Magnetic Resonance Imaging, etc. Mammography is a broadly utilized procedure for recognizing breast cancer which depends on analyzing Mammogram images that include masses and calcifications. The tumors are generally classified as benign if the shape of the tumor on the mass is rounded, smooth and transparent. Also, the benign tumors are categorized based on calcification shapes which are granular, coarser, ring or pop-corn shape and which has a higher density and a more distributed dispersion. The tumors are classified as malignant if the shape on the mass is needle-like with uneven edges and often fuzzy. Calcification has a morphology that is mainly sand-like, linear or branched, with various forms and sizes, and the distribution is usually dense or packed in a linear pattern (Li et al., 2019).

Even though Mammograms have been effective in early identification of breast cancer which has maximized the rate of survival in patients, external factors of error such as distractions, fatigue and manual errors while examining Mammograms need to be minimized.

The rate of false negatives of breast cancer during preliminary screenings of Mammograms is up to 30% high (Elter & Horsch, 2009). The interpretation of Mammogram involves complexity to differentiate between different cases as illustrated in Figure 1 which contains three different images related to normal, benign, and malignant forms which can be hard to find for an untrained human eye.

Figure 1

Mammograms of Normal, Benign, and Malignant cases



Note. The example shows Mammograms of different cases - Normal, Benign, and Malignant. Adapted from "*Breast cancer detection in mammograms using deep learning techniques",* by Jaamour et.al., 2020, *University of St Andrews - School of Computer Science*, p.2, (https://doi.org/10.5281/zenodo.3985051). Copyright 2020 by University of St Andrews.

Incorrect interpretation of Mammogram images by radiologists can lead to decisions that are harmful to the patients. In order to minimize such errors and improve the accuracy of Mammography screening, the use of Computer Aided Diagnosis/Detection (CAD) systems has proven to be effective. CAD systems are further classified based on the image classification

techniques involved, which are traditional CAD systems that are based on machine learning and modern CAD systems that are based on deep leaning.

The aim of this paper is to detect cases of breast cancer in mammograms using CNN with different classification models including AlexNet, ResNet-50 and then perform a comparative review on which model gives best results based on sensitivity, accuracy, and specificity. The primary challenge in breast cancer detection using CNN technique is that it requires a huge amount of data to train the model to attain great accuracy. To overcome this limitation, I have considered the data augmentation techniques to facilitate the required amount of data for CNN. The anticipated output from this study is to achieve better accuracy with less computational time, which would help improve the diagnosis of breast cancer with better performance and avoid false positives and negatives.

1.2 Project Requirements

This study is focused on identifying whether a tumor is malignant or benign using different classification techniques that involve Deep Learning. In order to fulfill the above-mentioned objective, gathering the data sources which are Mammographic images are quintessential. Mammograms are taken with a low dose X-ray imaging technique; the machine has compression paddles to hold the mammary glands and it could flatten or compress the breast to take images. Multiple images are taken, and they can be combined to form a 3D mammogram, which is a three-dimensional construction of breast. MIAS dataset which was taken from Mini Mammographic Database will be used for this project. The dataset is composed of Mammographic images which are usually in Portable Gray Map (PGM) format which are gray scale images with the lowest common denominator. The images in PGM format are further converted into Portable Network Graphics (PNG) format using Python Imaging Library. The

converted images still have strong noise which would require Data Preprocessing to reduce noise and further enhance image quality. The Data Augmentation is the key aspect of data preprocessing technique which aims at providing sufficient quantity of data for training the model. Following data preprocessing, data segmentation is performed to remove unwanted regions in the images.

Once the data is preprocessed and segmented, it will be stored in Amazon S3 cloud storage for further training and testing of the model through Amazon SageMaker. The data in the cloud storage is accessed by Amazon SageMaker where in different python libraries like SciPy, NumPy, Pandas, deep learning libraries like TensorFlow, Keras, etc., and machine learning libraries like Scikit-Learn are imported. The above libraries would provide for functional resources that would help optimize and evaluate the computational dataset with more accuracy. This would ensure that the output of the models is efficient and accurate. Once the models are evaluated based on accuracy, precision, recall, and F1-Score using confusion matrix and comparing their performance using AUC-ROC. The model with best performance is considered and deployed to AWS Sage Maker platform to make it available for consumption by respective medical professionals.

1.3 Project Deliverables

This project is developed and delivered in four major phases. The study starts initially with the analysis phase that identifies existing challenges to form a problem statement. The main problems are outlined in the literature review with information related to previously available breast cancer detection methods combined with different computational techniques. In the second phase, a workflow model of the project is developed to represent the schematic view of the process along with description of necessary functions like data collection, testing and

evaluation of the results. The workflow diagram will involve different stages of the process such as dataset selection and preprocessing of the data. The third phase comprises of implementation and testing of the model. Design plan and data from the previous steps are structured and aligned for implementation. The preprocessed data is fed to the selected models to train and evaluate. The fourth phase is focused on observation and evaluation of the data. The final deliverable of this project involves reporting the proposed deep learning pipeline that helps improving the performance of breast cancer detection.

1.4 Technology and Solution Survey

The most common screening method to detect breast cancer is using mammograms which require conventional diagnosis of professional radiologists. Pre-1990s CAD software has been utilized to aid radiologists in reading and understanding mammograms which were crude and did not provide much more information than skilled radiologists. In the late 1990s, supervised machine learning approaches began to replace traditional expert systems, allowing these new algorithms to discover hidden patterns in mammography data that radiologists could not see. As a result, we have seen a shift away from systems that are entirely created by people and towards systems that are educated by computers using example data (Litjens et al., 2017).

Machine learning algorithms that are commonly employed for breast cancer classification are Support Vector Machine (SVM), K-Nearest Neighbors (K-NN), Artificial Neural Networks (ANN), Random Forest (RF), and Logistic Regression (LR). Several people have made contributions to this subject and developed models to diagnose breast cancer in the past, and they have been effective in achieving exceptional outcomes with high accuracy and precision. ANN and SVM are two machine learning techniques that have proven to be effective in reaching high accuracy and precision and are widely used (MurtiRawat et al., 2020). The major drawback of

using these machine learning algorithms are they were unable to operate on raw data such as mammography images. To overcome the challenges this model require relevant features to be extracted from the image using image processing techniques.

Allowing computers to learn the features that best reflect the data for the situation at hand is a reasonable next step. Multiple deep learning models, which are completely fabricated of many layers and transform raw image content into classified outputs while learning increasingly higher-level features, are an example of this approach. However, these models have gained importance in recent times where they were being applied successfully since they need specific and powerful computing resources like systems with Graphical Processing Units (GPU) to train efficiently. CNNs have proven to be the most successful image processing model to date (Litjens et al., 2017). CNNs can now interpret images that are thousands of pixels in size, far faster than classic machine learning approaches. There are 100,000,00 connections exclusively in the first layer in a CNN with mere 100 neurons. In CNN for classification, different models are used like InceptionV3, DenseNet121, ResNet-50, VGG16, and MobileNetV2. Among them the Inception V3 and an altered U-Net model offers the best results with good accuracy, sensitivity, precision, and less computational time (Salama & Aly, 2021).

1.5 Literature Survey of Existing Research

Many novel approaches for detecting breast cancer have been created as medical science has progressed. The following is a brief summary of the research in this field.

Aslan et al. (2018) used routine blood analysis for detecting the breast cancer using different machine learning algorithms. For this study they used Breast Cancer Coimbra dataset which has features like age, glucose, insulin, body mass index, homeostasis model assessments, resistin, leptin etc... Four different machine learning algorithms ANN, SVM, standard Extreme

Learning Machine (ELM), and KNN were used for classifying the target data as healthy and unhealthy. By using the hyper parameter optimization methodology, the hyper parameter values producing the least errors for all the four models were obtained. These values were used to calculate accuracy rates and training times. Standard Extreme Learning Machine has the best accuracy rate of 83.87% and the shortest training period compared to other three models. They demonstrated that when there are large number of samples, using a Standard Extreme Learning Machine (ELM) saves time.

Omondiagbe et al. (2019) offered a fusion strategy to identify breast tumor that involved utilizing Linear Discriminant Analysis (LDA) to trim the high feature dimensionality to apply the new trimmed feature dataset to different ML algorithms to classify benign and malignant tumors. The ML algorithms like SVM, ANN, and Naïve Bayes (NB) are used for classification. According to the findings Support Vector Machine-Linear Discriminant Analysis (SVM-LDA) and Artificial Neural Networks-Linear Discriminant Analysis (ANN-LDA) together had excellent results with a sensitivity of 98.4%, precision of 98.4%, and accuracy of 98.82 %.

Although the above two approaches gave better results, the use of deep learning techniques would significantly reduce the Image pre-processing steps for feature extraction which cannot be avoided in case of machine learning. According to Shen et al. (2019) end-to-end deep learning models can be extremely accurate and potentially adaptable across a variety of mammography platforms. As accessible training datasets and computer resources grow, deep learning methodologies offer a wide scope to increase the accuracy of diagnosis of breast cancer via screening mammography.

Salma and Aly (2021) proposed a fresh model for breast cancer image classification and segmentation based on Mediolateral Oblique (MLO) and Cranio Caudal (CC) views to improve

system performance. Breast cancer is often detected and diagnosed using the CC view and MLO view. As the number of views grow, the accuracy of breast cancer detection will improve. This study used three mammographic datasets which are Curated Breast Imaging Subset of DDSM (CBIS-DDSM), MIAS, and Digital Database for Screening Mammography (DDSM). In order to classify the data into malignant data or benign data, a variety of practices like VGG16, DenseNet121, MobileNetV2, InceptionV3 and ResNet50 are used. This paper introduces completely CNNs from end to end. With the DDSM dataset, the proposed approach with InceptionV3 and an altered U-Net model yields the results with 98.870% accuracy, 98.980% sensitivity, 97.990% F1 score, 98.790% precision, and 1.21340s computing time.

Alanazi et al. (2021) proposes a CNN method to significantly improve the automated early detection of breast cancer based on the analysis of affected ductal carcinoma tissue regions in Whole-Slide Images (WSIs). All the architectures were driven by 50 x 50-pixel RGB image patches of large datasets which are typically about 2,75,000 in number. The validation tests were performed for quantitative results based on the concerning performance measures of respective methodologies. Proposed CNN Model is comprised of two convolution layers each of which has 32 kernels and 64 kernels respectively. The convolutional layers are kept in check with a dropout regularization to prevent overfitting. Vectorization of the image is done with a flattened layer to produce next dense layer. Rectified Linear Unit (ReLu) that is applied across all the layers but the output layer, where the SoftMax activation
function is used. The proposed system is found to be effective and successful with the achievement of results that are 87% accurate.

All the above-mentioned approaches aim for early detection of breast cancer with better accuracy with the respective data samples. From the literature survey it is evident that the

volume of data plays a crucial role in determining accuracy and efficacy of the detection models involved. Although the computing time is dependent on different parameters considered, the data preprocessing steps like augmentation which provides for sufficient amount of data to train the model and segmentation which improves the quality of data have complemented the deep learning models to achieve better performance with reduced computing time.

2. Data and Project Management Plan

2.1 Data Management Plan

Data collection is the key for breast cancer detection using Deep Learning methods. Different kinds of detection techniques use different methods of data collection like images from Mammograms, CT scan, MRI, Ultrasound etc. We have considered data from Mammographic images for this research project for detection of breast cancer based on the analysis of masses and calcification from the collected image data. Mammographic images are collected using X-Rays where the image collection machine holds the mammary glands of the breast in different positions to capture three-dimensional aspect of breast.

The original image data that is used for this research project has already been collected and it is complemented with associated truth data markings. The database is comprised of 322 images that are in digital format, which is originally existing on 8mm Exabyte tape with a size of 2.3 GB. The source of this digitized data is UK National Breast Screening Program where the images are reduced from 50 to 200-micron pixel edge with necessary clippings/paddings for 1024 x 1024 pixels of similar size and made available in PGM format (Suckling et al., 2015).

The metadata information consists of authors, data description along with the timeline of dates, format of the data used, copyrights etc. This metadata information is stored in a text file and uploaded to the root folder in Amazon S3 cloud storage. The metadata information in the root folder represents the original Mammographic data being considered. Standard file naming convention will be used to name the folders for different stages of image processing like data_augmentation, data_segmentation etc. The metadata of the indexed files in the respective folders will be stored in a directory structure parallel to the indexed files by using S3 prefix for the metadata files.

This research project involves real data collected from test subjects or patients at UK National Breast Screening Program which will have the image data along with sensitive information related to the patients involved. However, the extracted data in MIAS dataset contains only anonymous and open-source information which does not trace back to the patients involved in the diagnosis. Since the dataset used in this project is excerpted from mini-MIAS database of Mammograms, the copyright and legal terms and conditions follow the license agreement of MIAS.

The data is initially downloaded from MIAS Database into local machine. Then Amazon Command Line Interface (CLI) is installed into the local machine to establish communication and control AWS services. The next step would be to create a bucket in Amazon S3 Storage and inside the bucket a folder is created to sync the raw image data available in local machine. There will be two more folders created to hold the processed images after Data Augmentation and Data Segmentation.

In order to deal with any downtime or data loss due to network or infrastructure issues related to the S3 Storage, the S3 bucket will be safeguarded with S3 Cross Region Replication (CRR). Enabling this feature will allow the user to replicate the entire S3 Bucket to another region which provides the user with high availability and durability for the data. The source and destination would be located in different regions for CRR to be implemented. For instance, if the source bucket is located in AWS US West region, the replica bucket needs to be in AWS US East region. Amazon S3 Glacier is used for long term storage.

Access and security of the image data is governed and regulated by the Identity and Access Management (IAM) policies. Amazon S3 buckets are restricted to private use of the user by default and any other type of access to the bucket resources is regulated by Role-Based

Access Control (RBAC) via Access Control Lists (ACLs) (*Introduction to Amazon S3, n.d.*). As the image data stored in S3 bucket needs to be shared with Amazon SageMaker for processing of data, an IAM role needs to be created with Amazon SageMaker Full Access policy.

The image data in the S3 bucket under Preprocessed folder is shared with the Amazon SageMaker to build, train, and deploy the machine learning model. There is a hosted solution in SageMaker called Jupyter Notebook which facilitates the platform to write the code required to create model training, deployment of model and run any validation tests of the model. The data sharing and usage mechanism involved in this project is as below:

- Mammographic Image data downloaded from MIAS Database.
- MIAS Dataset is uploaded to Amazon S3.
- MIAS Dataset is shared from Amazon S3 to Amazon SageMaker for Data Augmentation.
- Data Augmentation results are then stored in Amazon S3 bucket in a data_augmentation folder.
- The augmented data is then processed for Image Segmentation.
- The segmented data is then stored in Amazon S3 bucket in data_segmentation folder.
- Segmented data is further used to build, train, validate and deploy the models.

Amazon Virtual Private Cloud (VPC) hosts all the resources like Amazon S3, Amazon SageMaker etc., and is responsible to provide designated access between the resources without the need to setup any clusters and pipelines for data flow.

Since the data in this project across different stages is shared across internal systems only, public access is restricted using Amazon S3 Block Public Access feature which blocks the centralized public usage. This research project is conducted at the researcher's sole discretion who will be responsible for the management of data bound to the license agreement of MIAS.

2.2 Project Development Methodology

This undertaking depends on Cross-Industry Standard Process for Data Mining (CRISP-DM) process model which is based on the six phases of life cycle for data science as explained below.

2.2.1 Business Understanding

In Business understanding, the first step to do is to view the problem in detail, which in our case is Breast cancer. Breast cancer growth is considered as one of the main sources for worry in ladies which takes a lot of lives each year. Early detection and diagnosis can be lifesaving and also can avoid the progressive damage done to body tissues. Misinterpretation of mammograms by radiologists can put the patients in danger. To avoid or to reduce these errors and to better detect the tumors CAD system is used. Machine learning can be effective in predicting breast tumors with a great accuracy. The downside here is ML techniques cannot operate on raw mammography images. We need a new solution that can overcome these challenges in existing methodologies and more accurately detect Breast cancer. The main goal of this paper is to propose a new optimized deep learning pipeline, that can work with raw image data and can produce accurate detections. To achieve this, we will design a project plan that can give us an 80% to 90% predictive accuracy, execute it, and measure its progress.

2.2.2 Data Understanding

Understanding the data is a crucial task in any project that deals with raw data. The MIAS dataset contains gray scale mammographic images with the lowest common denominator. There are labels and annotations for mammography scans and a total of 322 images were present in this dataset. Three different image classes benign, malignant, and normal are in that dataset. All images are 1024x1024 pixels and have been centered in matrix. Now we perform a check to

see if the image dataset is sufficient to train our models. We need to look over the data thoroughly to find if all the images are of same dimension and if there is any corrupt or black only image and remove them from the dataset. It is essential that we avoid data imbalance while selecting data. Looking at the data, it is clear that these 322 images will not be sufficient to train our model, so we will perform certain augmentation techniques in data preparation phase that improve the data quantity.

2.2.3 Data preparation

Mammogram is a high-resolution image, but it is hard to assess these images with naked eye. Hence, there is a need to do preprocessing. First start by removing the noise from raw image dataset which would otherwise lead to many inaccuracies. These noises are high and low intensity rectangular label and tape artifacts. Then, we would separate the tumor area from background by delineating it, extract the breast boarders and suppress the pectoral muscles. Data augmentation is used to create new data samples, this avoids overfitting by enhancing generalization capacity of pre-trained model and also speeds the converging process. It introduces variety to the dataset as transformations like rotation, scaling, flips, shear, and translation are applied to each image. To further process data, data segmentation will be done. This would isolate the selected regions of interest breast region from background which are then masked with raw mammogram data. The resultant data is segregated into training, testing, and validation.

2.2.4 Modeling

Data preparation is followed by selecting suitable models to classify data into benign, malignant, or normal. To do this, we choose modified CNN and pre-trained models such as AlexNet, and ResNet-50. CNN model training takes a lot of computing power and can take

thousands of hours. Softmax activation functions, and multi-class cross entropy loss function will be used. Then, build the chosen models. Adaptive Adam optimizer is used to reduce the number of hyperparameters for better control. Adam optimizer contains benefits of Root mean square propagation and Adaptive gradient algorithm. As the dataset is small in size, it could cause data imbalances, sampling bias, and to counter this we stratify the splits. This will give us an 72% training, 20% testing and 8% validation datasets and using these datasets we train and test the chosen models using python TensorFlow libraries. And using validation set predictions are made considering loss and accuracy at each epoch end.

2.2.5 Evaluation

There are a lot of evaluation methods, most of them show no indication of their performance in predicting data that is new to them. So, to improve the accuracy of prediction, we will not use the complete dataset to train the model. The Performance of the model is tested by the method called cross validation which is performed by using the set of data that is removed before training the model. We prefer k-fold cross validation to make predictions because it works best with a limited dataset and makes sure that initial dataset has a probability of making it to training and test dataset. And the above selected models are evaluated using the metrics such as accuracy, recall, precision, F-Score and AUC-ROC curve. After calculating the metrics for all three models, the outputs, performance, and execution time is compared with one another. The error in prediction for all the three models is analyzed, and if necessary, remodeling is done. Finally, the best model will be chosen for deployment.

2.2.6 Deployment

An effective strategic plan for deployment is crucial for every project and it constitutes the final stage of the project. After evaluating all the selected models, the model that performs best in the testing environment is selected. This selected optimum model is deployed into production. The model needs to be monitored during and after the deployment for any errors that might arise. After the model is trained and deployed, proper maintenance support is also necessary for it to function as intended. The user feedback is required to further improve the quality of the model.

2.3 Project Organization Plan

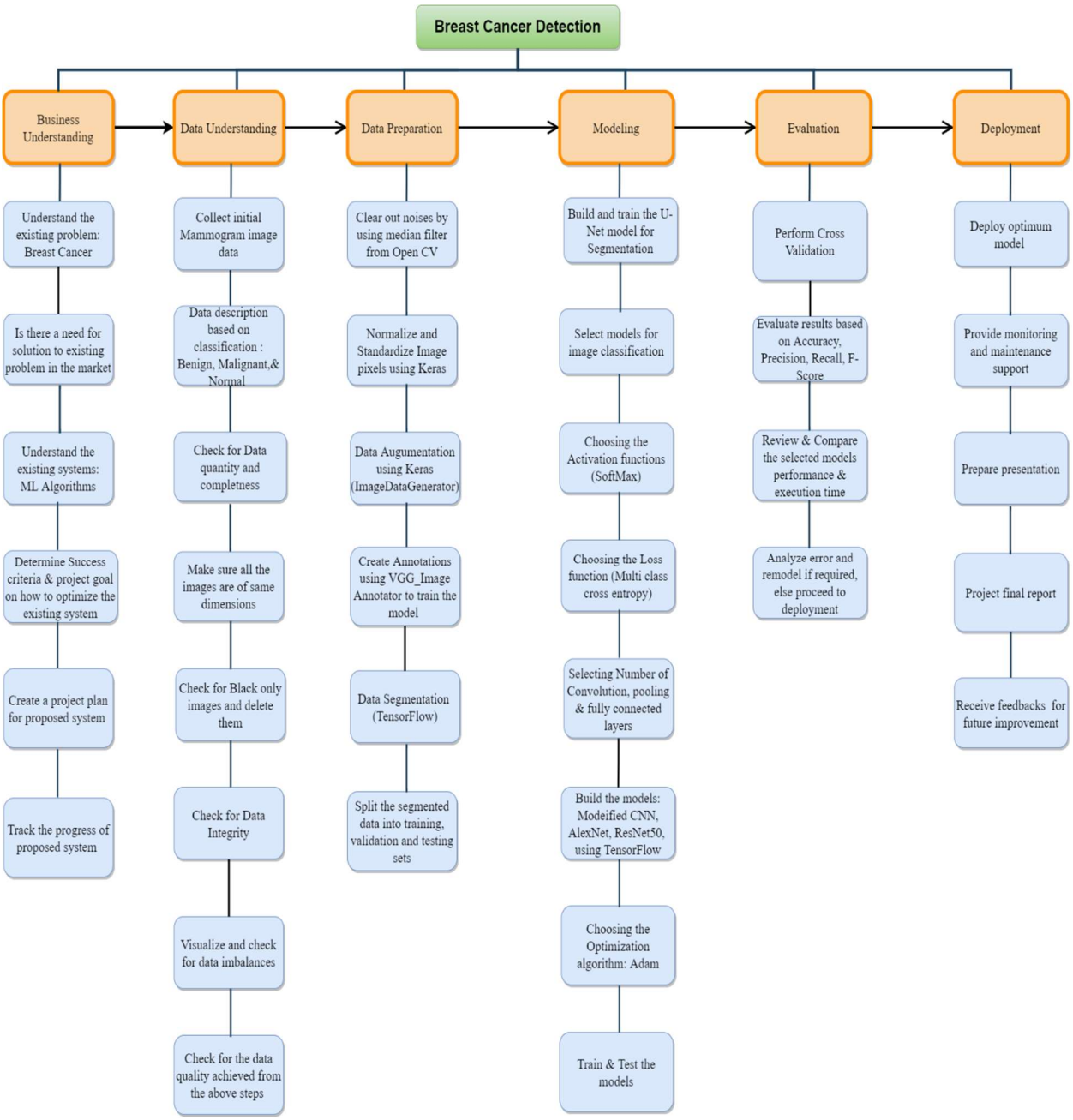
This project adheres to work breakdown structure as shown in Figure 2 to make sure the deliverables and the scope are aligned in a phased manner. The work breakdown structure is based on the CRISP-DM process model which ensures proper planning, execution, monitoring and control. This will also serve as a source of truth for different phases of this project execution. The CRISP-DM process model provides a road map for different phases involved in this project execution plan.

The Business Understanding phase first phase which is based on understanding the existing challenges with the breast cancer detection techniques in place and their associated algorithms. Laying out a project plan with success criteria and the details pertaining to optimization of existing methods with the measures to track the progress of the proposed methods will complete the Business Understanding phase. Data Understanding phase comprises of different steps. Data collection step lists the source of the data along with the technique used to collect the data and associated challenges if any that are encountered during data collection. The data description and data exploration steps involve explanation of the data source in detail

and analysis of the test data collected. The final step in Data Understanding is data quality which ensures the data integrity by making sure the irregularities in the data are excluded and qualified data with completeness is considered for next steps of data processing. Data preparation phase involves pre-processing methods to generate the data to be fed to the modeling phase. The rationale for preparing the data consists of actions that are needed to improve the quality of data by using techniques like data augmentation, data segmentation etc. Modeling is the fourth phase of Work Breakdown Structure that describes the assumptions of the data based on different modeling types used and the corresponding data that is to be trained. This phase also includes model optimization algorithms and testing plan. Evaluation phase is the fifth phase that is used to calculate the accuracy and assess the performance between selected models. If any errors or imperfections are observed, then a remodeling step is required else the optimum model is proceeded to deployment phase. The Deployment is the final phase that consists of deploying the optimum model, monitoring and maintenance plan for the deployed results and the finalized report which summarizes the results.

Figure 2

Work Breakdown Structure



Note. Work breakdown structure with hierarchical structure of different phases in the project.

2.4 Project Resource Requirements and Plan

There are different types of resources that are considered optimal for executing this research project. Table 1 depicts information about the functional resources associated with the project and their cost estimations.

Amazon VPC serves as a platform to orchestrate and integrate different hardware and software resources that form the building blocks of this project. With the advent of cloud hosted solutions like cloud computing and cloud storage it has become easy, effective, and economical to setup different resources required for a project like this. As there are multiple hardware and software resources that fall under the category of AWS, a unified tool is required to manage the different services. The AWS CLI is the centralized management tool that is required to be downloaded and configured in the local machine. This tool provides the functionality to communicate with different AWS resources and services using the command window in the local machine (*AWS Command Line Interface, n.d.*).

Storage stands out as one of the primary components for any project that involves software and hardware. This project is solely based on cloud storage services from Amazon. The storage types used in this project are classified into three types based on their purpose. Amazon S3 storage serves as the primary storage required to house the Mammographic images that are needed to be processed for analysis. Also, different processing stages utilize Amazon S3 to store the processed data and continue further by using the stored data to feed to the training model. Amazon S3 charges $0.023/GB per month which gives $2.07 for 30 GB for three months (*AWS Pricing Calculator, n.d.*). Amazon S3 Replication acts as a Disaster Recovery (DR) mechanism and would also ensure high availability in case of any contingencies. Usually, this alternate storage is stationed and hosted from an alternate physical location that is located at a distant

location from the primary storage. This type of DR architecture is termed as S3 Cross-Region Replication (*Amazon S3 Replication, n.d.*). Amazon S3 Replication costs involve time control data transfer cost of $0.015 per GB along with S3 destination storage charges of $0.023/GB per month which accounts for a total cost of $2.52 for three months (*AWS Pricing Calculator, n.d.*). Amazon S3 Glacier is the storage type that fulfills the archival needs of the data involved in this project (*Introduction to Amazon S3 Glacier, n.d.*). Archival data typically is stored for a longer life cycle without much need to retrieve. S3 Glacier costs $0.004 per GB per month which sums up to $0.36 for three months (*AWS Pricing Calculator, n.d.*).

Machine learning or deep learning models require an efficient and effective integrated development environment (IDE). Any machine or deep learning model typically consists of three major steps: build, train and deploy. Amazon SageMaker is identified as the best choice of IDE for execution of the workflow because it has integrated set of tools for process automation, error correction and it is also cost effective. It has built in capabilities for different machine learning components that are packaged as a tool set. Also, Amazon SageMaker is a flexible, scalable, and powerful IDE which can be scaled horizontally and vertically as required. Since SageMaker is also an integrated part of AWS, the date used or generated during various phases of the workflow like testing, training or validation can be stored in a common data storage. This IDE contains notebooks for running Python kernels with a customized compute instances that we can choose as per our requirement. Since this research project is based on building four different deep learning models, Amazon SageMaker ml.p3.2xlarge instance is required to meet the data modelling aspects (*Overview of Amazon EC2 P3 Instances, n.d.*).

Amazon EC2 P3 instances are known for high performance computational delivery that has significantly brought down the training times from days to minutes while iterating through

number of simulations at increased rates. The p3.2xlarge instance is complemented with one Tesla V100 Graphical Processing Unit (GPU), 16 GB GPU memory, 8vCPUs, 61 GB Memory and up to ten Gbps of bandwidth. Sagemaker with ml.p3.2xlarge instance costs $3.825 per GPU per hour which accounts for $153 (*AWS Pricing Calculator, n.d.*) for forty hours which is the estimated duration required for this project (*Overview of Amazon EC2 P3 Instances, n.d.*).

Python is used as programming language to orchestrate the infrastructure and define data processing and modelling. TensorFlow and Keras are the machine learning frameworks that are utilized to build, train, test and evaluate the models. OpenCV, NumPy, Pandas, Matplotlib and ImageDataGenerator are some of the noticeable libraries used for this project. Draw.io is an online diagram tool that is used to create the work breakdown structure, PERT chart and different diagrams and process flow charts in this project. The cost of usage for this online drawing tools is $15 per month which costs a total of $45 for the duration of this project (*Pricing, 2021*). The total cost estimate for entire project will be around $ 205.

Table 1

Resource and Cost Estimation

Function	Resource Type	Resource	Time Duration	Cost Estimation
Cloud Service Management Tool	Software	Amazon CLI	10/12/2021-12/03/2021 (3 months)	Free
Data Storage	Hardware	Amazon S3	10/12/2021 - 12/03/2021 (3 months)	$2.07
Disaster Recovery	Hardware	Amazon S3 Replication	10/12/2021 - 12/03/2021 (3 months)	$2.52
Data Archival	Hardware	Amazon S3 Glacier	10/12/2021 - 12/03/2021 (3 months)	$0.36
Build, train, test and deploy	Software	Amazon SageMaker ml.p3.2xlarge instance	10/06/2021 - 12/03/2021 (2 months)	$153
ML Frameworks	Software	TensorFlow & Keras	10/12/2021 - 12/03/2021 (3 months)	Free
Software Development	Software	Python 3.9	10/12/2021 - 12/03/2021 (3 months)	Free
Diagram Tool	Software	Draw.io	10/12/2021 - 12/03/2021 (3 months)	$45
Image Annotation	Software	VGG Image Annotator	10/12/2021 - 12/03/2021 (3 months)	Free

Note. Functional resources associated with the project and their cost estimations.

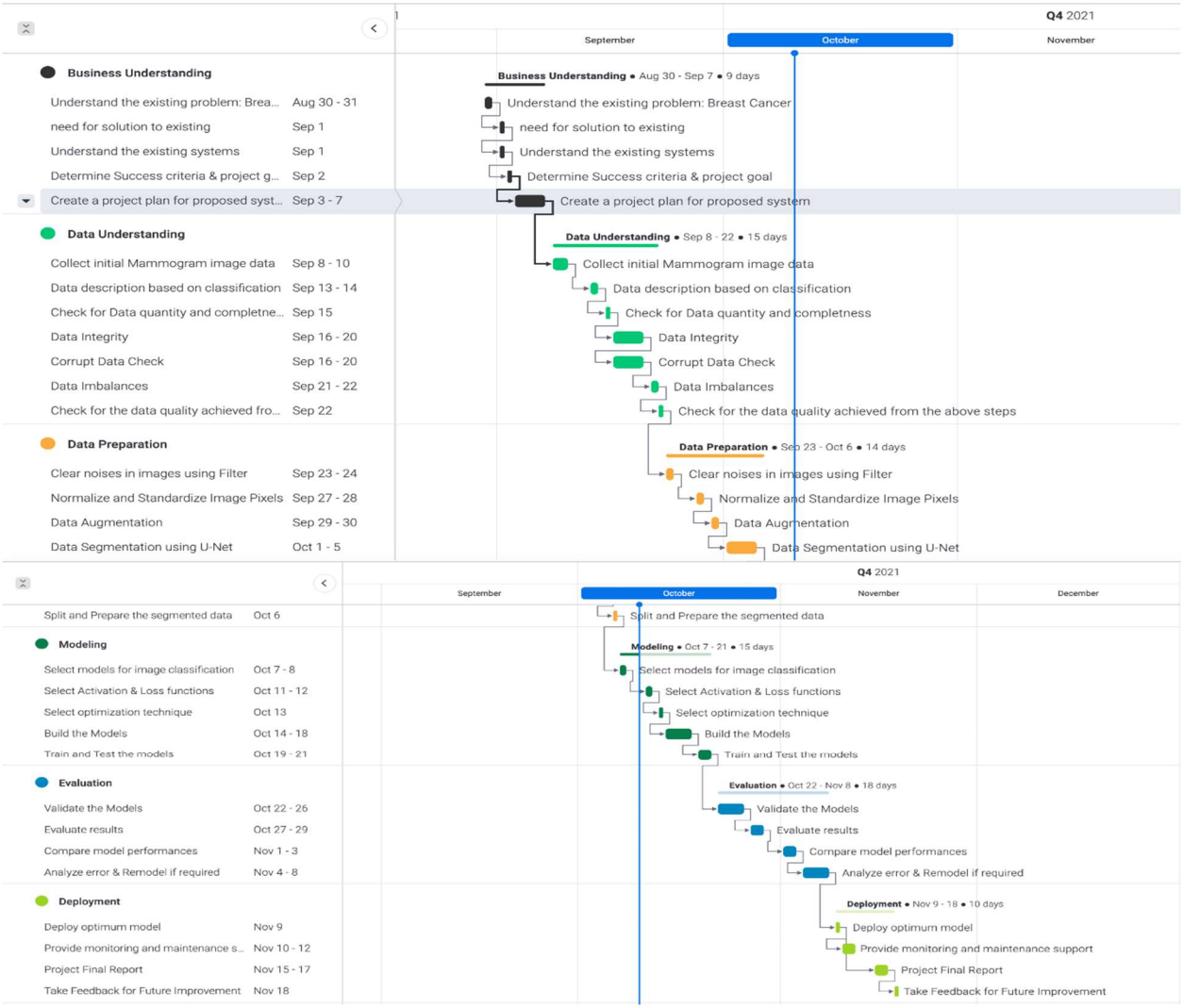
2.5 Project Schedule

Gantt chart is used as a tool to better visualize the project tasks distributed over designated time schedule for this project. Figure 3 Gantt chart is represented by a bar chart in horizontal orientation. The tasks are listed on the vertical axis in different rows and the time frame is depicted on the horizontal axis. The task list, timeline, milestones, and task dependencies are the crucial elements that are required to construct a Gantt chart. In this project, finish to start task dependency method is used which is represented by directional arrows

connecting each task. The direction of arrows indicate that the current task must be completed before the next task begins.

Figure 3

Gantt Chart



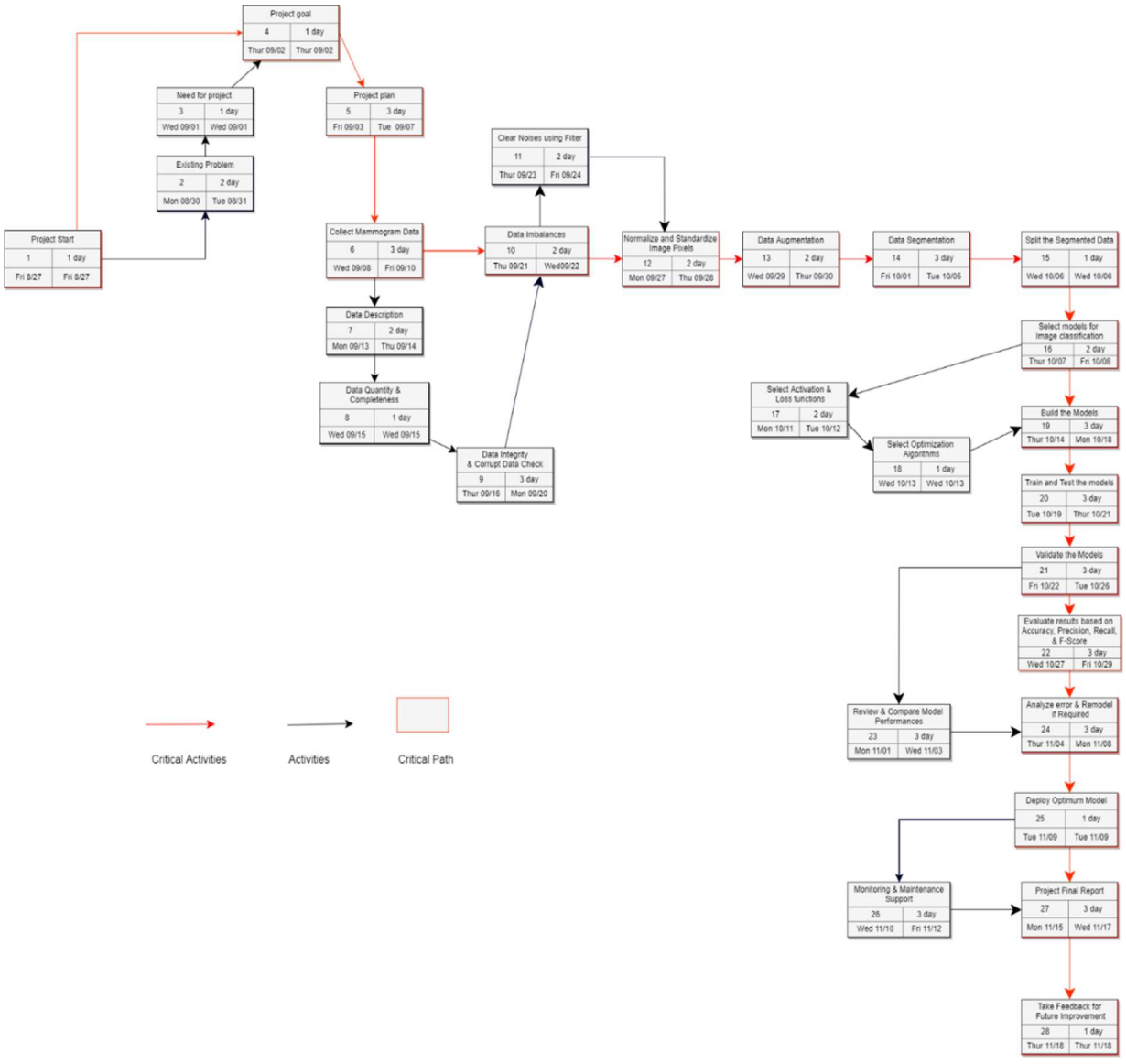
Note. Gantt chart tasks and timelines.

PERT chart is a tool that is used to analyze and estimate the effort required for fulfillment of different tasks in the project. The graphical template in Figure 4 is used to illustrate milestones and tasks of the project in the form of free flow diagram. PERT charts use directional arrows to define the sequential execution of tasks that are interdependent. The numbered boxes are nodes

in the diagram represent different tasks that are linked by black or red lines. Critical path analysis of the project is represented in the PERT chart with red lines and arrows to identify the tasks that are quintessential to complete the project. The concurrent arrows indicate that the tasks are dependent and must be completed in the sequential order.

Figure 4

PERT Chart



Note. PERT chart to identify critical activities and dependencies.

3. Data Engineering

3.1 Data Process

The primary focus of this project is to improve efficiency of early breast cancer detection techniques which could save a lot of women. Since this project is based on the data collected from mammographic images used in the development of breast cancer detection model, the objective of data collection is focused on how the images are captured and collected from different human subjects. Any data collection effort involving humans will have to undergo a series of regulations and compliances to avoid any unforeseen issues related to data misuse.

The first level of approval for this research project is required at the university level, where the researcher must verify from the Institutional Review Board (IRB) exclusion worksheet if this work qualifies for the definition of research that involves human subjects and their individually identifiable private information. Since this project collects the identifiable information from the patients, it is evident that an IRB approval is required. The request would then undergo a structured review of protocols by the IRB reviewers where they would ask for prevalent information related to the research and any associated data related to it. The researcher will be able to provide all the required information related to the specifics of the project like introduction, problem statement, objectives of the research, challenges with existing methods or practices etc. The privacy of the subjects considered for collection of the mammogram images and the confidentiality of their information is another key aspect under IRB review, where applying for a certificate of confidentiality might be required based on the risks associated with respect to the subjects, how sensitive is the information collected and precautions taken to handle the collected information and protection of data privacy *(Humans Subjects Research - Institutional Review Board (IRB) | Office of Research, n.d.)*.

The consent form is another important aspect which is carefully examined during the review, which is required to fulfil the general requirements laid out by federal rule in 45 CFR 46 section 116. It is also required to comply with the combined consent requirements of Health and Human Services department and SJSU to pass the review. After all the forms and documents related to IRB review are filled with the required information, the faculty or the supervising staff member need to undergo a preliminary training course provided by SJSU. Although the training course is not mandatory for students conducting the research, it is advised that the researcher conducting the project that involves human subject in the research should take up the Collaborative Institutional Training Initiative (CITI). After the submitted information has been carefully reviewed and approved by IRB, the researcher will proceed with the data collection process *(Humans Subjects Research - Institutional Review Board (IRB) | Office of Research, n.d.)*.

Data process provides high level overview of different steps of operations like data collection, pre-processing, data augmentation, data segmentation etc., performed on the data at different stages of the project. The motive for processing the data is to convert the data from a given raw form to much more effective and desired form that is relevant and informative for the model. Understanding the different types of data used among different detection techniques forms the basis of selecting the data that is needed for this project. Among different techniques that are currently in use like Mammography, CT scan, Breast Ultrasound etc., breast cancer detection using mammography technique is widely used and cost-effective (Michael et al., 2021). The data that is being used in this project consists of mammographic images which are excerpted from MIAS Mini Database. The raw image data collected is in PGM format which are gray scale images. The initial step in data process consists of converting the images in PGM

format into PNG using Python Imaging Library. Next step would be to check for the data quality that involves checking if the images from the dataset are of uniform dimension, identify corrupt or images with missing attributes and exclude them if detected. There are three types of image classifications in the dataset namely benign, malignant, and normal breast images. The next step is to check for any data imbalances to ensure that the image data is not biased to a particular type. If any imbalances are observed, they are eliminated by using under sampling or over sampling techniques as required. Noise is an unavoidable feature that is associated with the mammogram images during acquisition. It is a challenging task to denoise the images while restoring the important aspects of the image. The different noises that affect the mammogram images are poisson noise, speckle, salt and pepper etc., (Devakumari & Punithavathi, 2018). Different filters like median filter, mean filter etc., are employed to reduce the noise in the images (Devakumari & Punithavathi, 2018). Based on the detected noise type, appropriate filters are used to denoise the mammogram images.

The size of data set has a significant effect on determining the quality of outcome from the model. The current data set comprises of 322 mammographic images of breast. However, this is considered a limited data when it comes to processing through deep learning models. Less amount of data results in bad performance due to problems like overfitting of training data that could eventually lead to improper prediction. To overcome this challenge data augmentation technique is used to generate more samples of data by using basic image manipulations like scaling, flipping, rotation of images to varying degrees etc. This technique will increase the diversity of test data and provide the model with sufficient data.

The next step in the data process is data segmentation which is crucial in detection of breast cancer and further provide valuable information in treatment planning. Detection of breast

cancer with proper diagnosis at early stage not only improves the type of treatment but also can prevent deaths. The detection of tumor is dependent on how accurate the breast region of interest is segmented. Although there are different image segmentation models available, U-Net model is the widely and frequently used segmentation technique that has proven to be effective in deep learning as it is not much dependent on annotation of the image data (Neha S, 2020). The image segmentation process will provide the output images with information containing region of interest while masking the unwanted regions from the pre-processed images. The image data obtained from all the above image processing and transformation techniques is then classified into data sets for training, testing and validation.

3.2 Data Collection

Collection of a well explained dataset is the basis for construction of an efficient model for any machine learning project. Since this project is based on the secondary data collected from MIAS dataset of mammographic images to develop breast cancer detection model, the objective of data collection is focused on how the images are captured and collected.

Mammography machine is a specialized medical imaging device which operates with x-rays that are low in radiation. It consists of a tube inside a box for production of x-rays. The background behind the use of x-rays is based on the fact, that x-rays travel through most of the objects including human body. The behavior of x-rays with human body varies in degree with density. Dense masses like bone structures absorb more radiation resulting in white images on the x-ray and soft masses like muscle, organs etc., are represented by different shades of gray while air passages are denoted by black area. The technician involved in capturing the mammogram images uses low radiation x-rays to direct towards the breast area from which the scanned images need to be collected. The Mammography unit also consists of a special

accessory called compression paddles with two plates between which the breast is positioned, compressed, and flattened to spread the region of breast tissue apart. This ensures the low x-ray radiation to pass through the tissue and capture a better picture of the masses inside the breast (Acr, 2021).

In mammography low intensity x-ray procedure is employed to visually navigate internal structure of breast of the patient. Each of the breasts are scanned with two different orientations which are from one side to other side and from top to bottom of the breast. When the screened image is observed opaque and white area denotes breast tissue and semitransparent and darker area denotes fatty tissue. Traditionally x-ray images were printed on photographic films, but with the advent of digital imaging techniques the x-ray images are now digitally recorded which provides a hassle-free storage and retrieval options. Mammogram images comprises of exposed and unexposed breast regions where the unexposed breast region is removed during preprocessing phase for generating the images with region of interest (Acr, 2021).

The primary goal of data collection is to ensure that the data collected is rich in information and image quality. Some of the baseline standards for image quality in mammographic images collected for breast cancer detection are blur free images that are high in detail, low noise, good contrast sensitivity etc.

The MIAS is an UK based National Breast Screening Program which is focused on understanding of mammograms for cancer ascertainment has produced a centralized database of digital mammographic images. The x-ray films from the breast screening program were carefully selected and digitized with an instrument called Joyce-Lobel microdensitometer for scanning of images where resolution of each image is adjusted by 50 μm x 50 μm and every pixel in the image is represented by eight encoded bits. The MIAS dataset contains a total of 322 images in

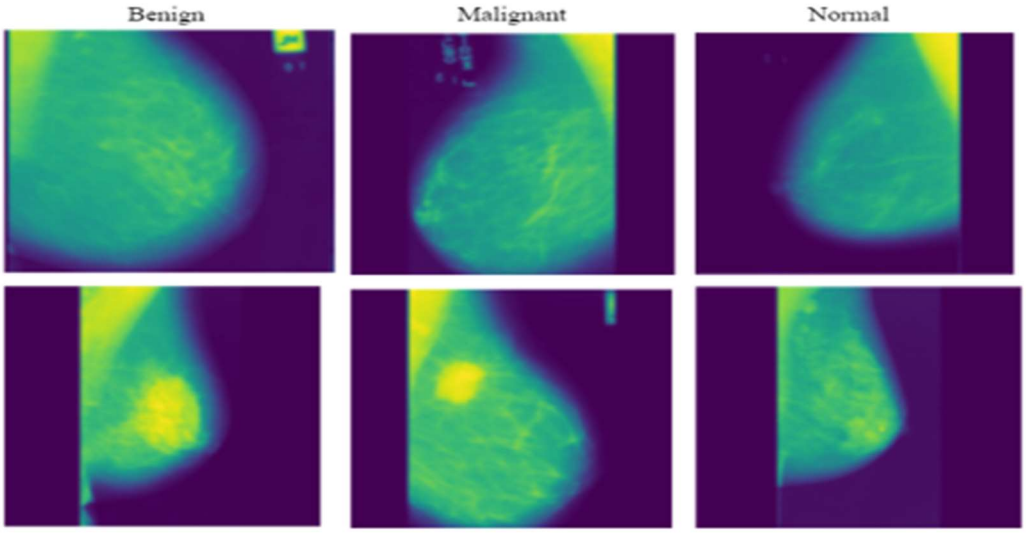
which 63 are benign, 208 are normal, and 51 are malignant that belong to both sides of breast sections of 161 patients. The screened images were broadly classified into three categories depending on how severe the abnormality is as shown in Figure 5 and Figure 6. The images collected in the MIAS Database were reduced from 50 μm pixel edge 200 μm pixel edge and resized for uniformity of 1024 x 1024 pixels for each image (Suckling et al., 2015).

MIAS dataset has a wide range of use in different areas of breast cancer research like segmentation, tissue classification of breast, detection of micro-calcification etc. The dataset has provided detailed information in the form of seven types of metadata related to the mammographic images, which are background tissue characteristics, types of abnormalities, severity of the tumor detected, coordinated location of micro-calcification and radius of circle surrounding the impacted region. The background tissue characteristics have three categories namely fatty which is denoted by letter F, fatty-glandular which is denoted by letter G and dense-glandular which is denoted by letter D as show in Figure 7. The abnormalities are classified into seven categories as described in Figure 8. Severity of tumors are denoted by letter B if it falls under benign category and letter M if it is categorized as malignant. Coordinates of center of impacted region and radius of impacted region were also provided for each image in the form of metadata. The filename of the images contains mdb as prefix followed by the image number (Suckling et al., 2015).

The metadata information as mentioned above is provided in a text file which will be used during pre-processing of data.

Figure 5

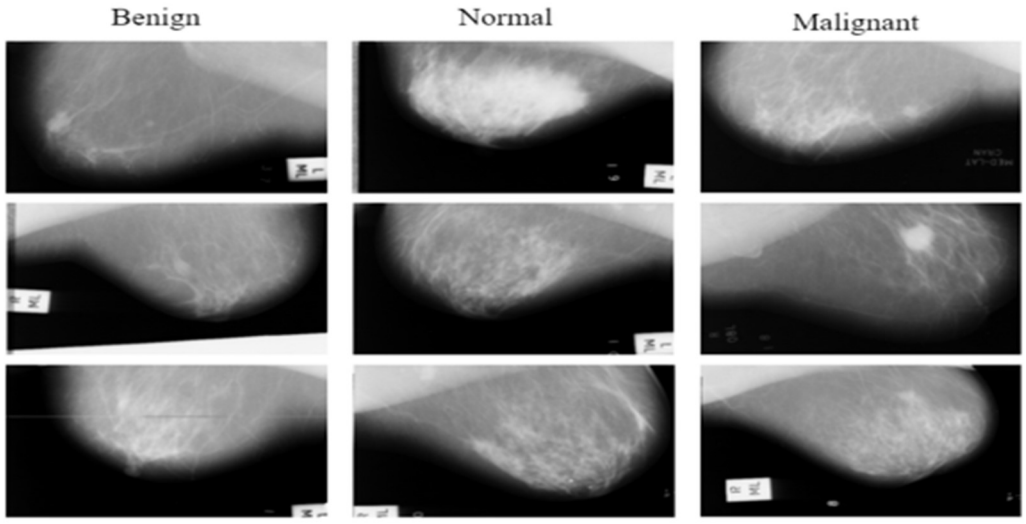
Samples of Benign, Malignant and Normal Breast Mammograms in PGM Format



Note. Samples of Benign, Malignant and Normal Breast Mammograms in PGM Format. Samples extracted from the mini-MIAS dataset (Suckling et al., 2015). Created using draw.io.

Figure 6

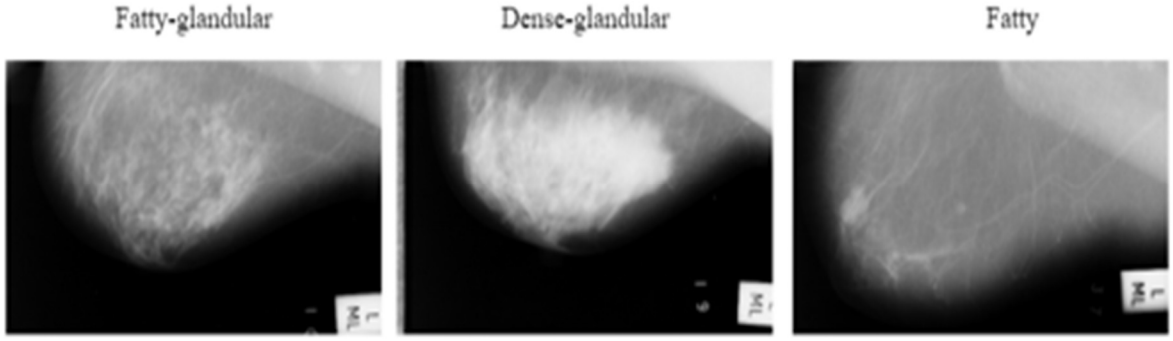
Samples of Benign, Malignant and Normal Breast Mammograms in PNG Format



Note. Samples of Benign, Malignant and Normal Breast Mammograms in PNG Format. Samples extracted from the mini-MIAS dataset (Suckling et al., 2015). Created using draw.io.

Figure 7

Samples of Fatty-Glandular, Dense-Glandular, and Fatty Breast Mammograms



Note. Samples of Fatty-Glandular, Dense-Glandular, and Fatty Breast Mammograms. Samples extracted from the mini-MIAS dataset (Suckling et al., 2015). Created using draw.io.

Figure 8

Samples of Truth Data Associated With Breast Mammograms

REFNUM	BG	CLASS	SEVERI	X	Y	RADIUS
mdb001	G	CIRC	B	535	425	197
mdb002	G	CIRC	B	522	280	69
mdb003	D	NORM				
mdb004	D	NORM				
mdb005	F	CIRC	B	477	133	30
mdb005	F	CIRC	B	500	168	26
mdb006	F	NORM				
mdb007	G	NORM				
mdb008	G	NORM				
mdb009	F	NORM				
mdb010	F	CIRC	B	525	425	33
mdb011	F	NORM				
mdb012	F	CIRC	B	471	458	40
mdb013	G	MISC	B	667	365	31
mdb014	G	NORM				
mdb015	G	CIRC	B	595	864	68
mdb016	G	NORM				

Note. Samples of truth data associated with Breast Mammograms. Samples extracted from the mini-MIAS dataset (Suckling et al., 2015).

3.3 Data Pre-processing

Data pre-processing is a type of data mining technique that is used to transform the raw data collected from different subjects or sources into a structured quality dataset that is well sorted and organized. Data collection process involves gathering of scanned images from different subjects which may not be screened in a well-controlled environment resulting in different combinations of data with inconsistencies. Missing metadata, low resolution images, distorted images, images with inconsistent aspect ratio etc., are some of the features affecting the quality of the collected data which could lead to complicated training phase and eventually produce results that are misleading.

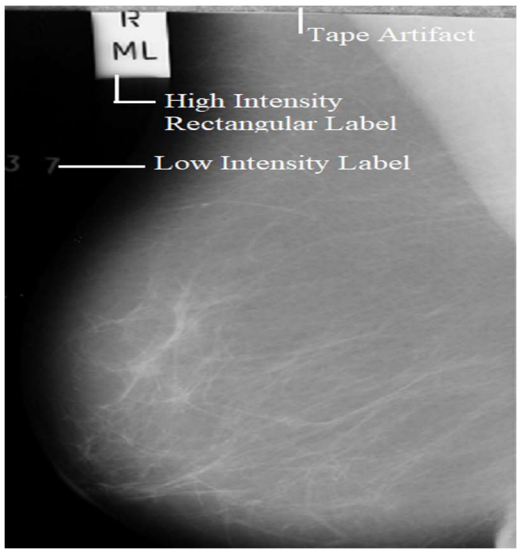
Data pre-processing plays a significant role in determining the efficiency of the model by providing with clean, unbiased, and non-redundant quality dataset. The raw image data collected is in PGM format which is incompatible with most of the image processing libraries and annotators. Since PNG format is supported by wide variety of systems and the conversion of PGM to PNG format will involve minimum compression without impacting the image quality or compression ratio, the raw images are converted to PNG format. Some of the mammographic images contain unwanted information lacking in abnormality regions. Removal of such images is a part of data pre-processing that is achieved by deletion from the dataset. After deleting the images which does not contain the region of abnormalities, the remainder of the dataset that is left for further processing accounts for a total of 319 images.

Enhancing image quality is achieved by removing unrelated and undesired parts that exist in the scanned images in the form of background noise. There are different noises that exist in the mammogram images like rectangular labels, tape artifacts etc., as shown in Figure 9 that are very difficult to interpret. Noises like tape artifacts and rectangular labels which exist in non-

significant area of the images can be eliminated by using cropping technique. Extracting border region of breast and suppression of pectoral muscle is also a part of preprocessing (Ponraj et al., 2011).

Figure 9

Noises Observed in Mammograms



Note. Noises observed in Mammograms. Sample extracted from the mini-MIAS dataset (Suckling et al., 2015).

Mammogram images are commonly affected by different noises like Gaussian noise, Speckle noise, Salt and Pepper noise, and Poison noise as shown in Figure 10. Noises appear in the form of small grains in the mammographic images that inflict fluctuations and thereby causing dissimilarities in values of image intensity. Salt and Pepper noise falls under the category of impulse noise where the values of the pixels in the image are lost and displaced by extreme values of the dynamic pixel value range of an image. For example, if we consider a gray scale image the values zero and 255 respectively are the pixel values with least and highest intensity. Typically, when the image is affected by this noise, the noise appears in the form of

white and black colored dots in the image. Gaussian noise is also addressed as amplifier noise which follows normal distribution. It is usually induced due to noise from electronic circuit or sensor. Poisson noise is also identified with the terms photon or shot noise. It appears in the mammographic images due to numerical change in measurement during screening process which is dependent on electromagnetic waves like gamma rays and x-rays. The cause of this noise is mainly due to the characteristic features of the measuring devices such as electrons and photons in electronic circuits or optical devices. Speckle noise is similar to gaussian noise. This noise is the resultant of granular noise which is caused by fluctuations in image signal waves that return from the breast during capturing (Devakumari & Punithavathi, 2018).

Figure 10

Example of Different Types of Noises

Note. An example of different types of noises. Adapted from "A Proposed Model for Denoising Breast Mammogram Images," by Hamed et al., 2018, *International Conference on Computer Engineering and Systems (ICCES),* p.3 (https://doi.org/10.1109/icces.2018.8639307). Copyright 2018 by ICCES.

The noises in the mammogram images are tackled using different filters and this process is called denoising. Different denoising methods are broadly classified based on type of filters used. Filters are employed to eliminate the noises and thereby improving the image quality. There are different filters like Median filter, Mean or Average filter, Gaussian filter, Noise

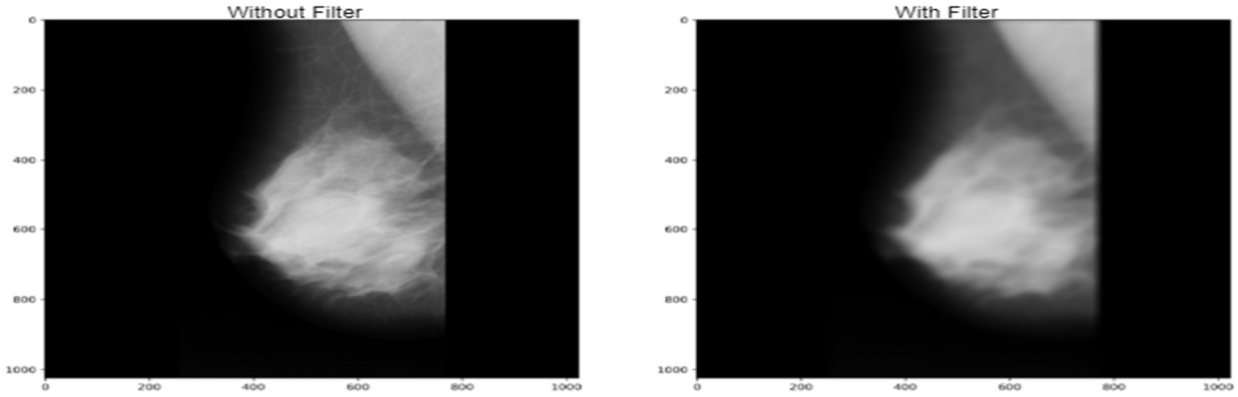
Adaptive Fuzzy Switching Median (NAFSM) filter etc. Median filter is also known as nonlinear filter employed to remove salt and pepper noise which is impulsive noise. In this filter technique, the pixels of the affected images are replaced by median rather than average of all the pixels. Mean filter on the other hand works by replacing each pixel in the impacted section of the image by average of intensity values in the surrounding area. The downside of using Mean filter is that it results in image blurring, and it doesn't remove the impulse noises completely but just reduce the impact. Gaussian filter is used to detect the peaks and perform noise correction of spectrum coefficients of the peaks detected in the given filter window. This filter is a linear low pass filter type that is responsible for smoothening of the pixel edges in a controllable manner (Devakumari & Punithavathi, 2018).

NAFSM filter or simply Salt and Pepper Noise reduction filter is a two staged filter that is used to detect and remove salt and pepper noise. The noise pixels in the histogram of impacted image are detected in the detection stage. The identified noise pixels are then extracted and eliminated in the filtering stage. The uncertainty during extraction of pixel information of noise induced pixels is handled by fuzzy logic which proved to be very effective among different filters for reducing salt and pepper noise (Devakumari & Punithavathi, 2018).

After examining different filter techniques that can handle the different noises in the mammographic images considered in this project, it is determined that median filter will better filter the different noises present in the images for enhancing the quality of the dataset. From Figure 11 as shown below it can be observed that the noise pixels have been replaced with the median pixels. The median filter from OpenCV library in Python is used to achieve noise reduction.

Figure 11

Mammogram Image Without and With Median Filter

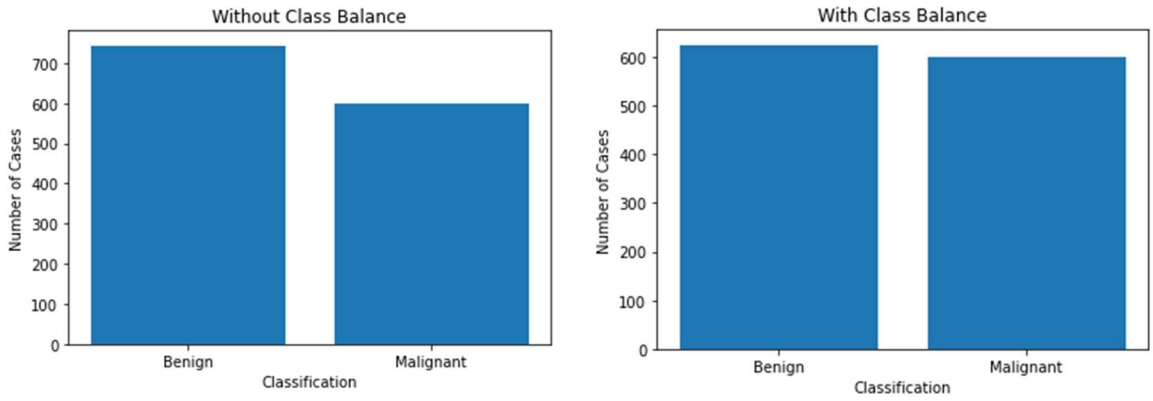


Note. Mammogram Image without and with median filter. Sample extracted from the mini-MIAS dataset (Suckling et al., 2015).

After exploring the data carefully and visualizing the dataset, it is observed that the data is highly imbalanced. Highly imbalanced data will lead to biased information among different classes or types to be analyzed from the dataset. When the imbalanced data is provided to the model, the model will lean towards the predominant class and might result in generalization errors. The model that is trained with an imbalanced data will prove to be futile and inaccurate with new data. To address the data imbalance issue, class balancing techniques are used where the dissimilarities between the classes or types of samples in the dataset are reduced by under sampling to achieve a balanced dataset. As illustrated in Figure 12 it can be observed that the calcification image samples for benign are more than malignant. Under sampling technique is used to balance the difference of samples.

Figure 12

Data Before and After Balancing Benign and Malignant Classes



Note. Data before and after balancing benign and malignant classes.

3.4 Data Transformation

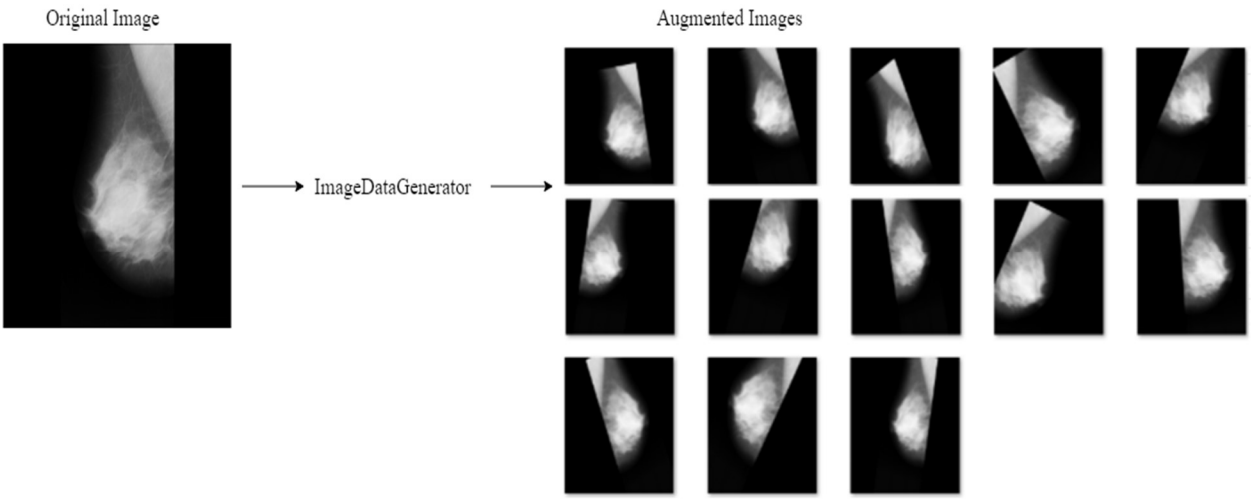
Deep learning models are reliant on large datasets of mammographic images for developing an efficient and well-trained model. Limited data availability for training usually results in overfitting problem which significantly impacts the accuracy of the model. Overfitting condition occurs when less data is provided to the training model. When less data is provided to train the model it will result in high accuracy, but the same model will result in less accuracy while testing since the model will not work accurately with the new data which it has not analyzed before. Smaller datasets in the analysis of medical images are a greater challenge and data augmentation will address the issue by providing enough data to train the model and reduce variance of the model which will improve the model performance.

Data Augmentation is aimed at generating more samples of the dataset for training the model. The additional data samples are created from the available datasets by using different image transformation and transition processes like randomly scaling the images, performing different rotations on the images, mirroring the images, flipping, increasing, or decreasing the

resolution, adding noises etc., (Tripathy & Swarnkar, 2020). When data augmentation is performed on the original dataset of 319 images, the initial step will generate subsamples in the form of mirrored and rotated images which are resized from 1024x1024 to 256x256 and rotated in varying degrees of 0, 90, 180 and 270. Image mirroring will include both top-bottom and left-right transformations of the actual images. The resultant of images after data augmentation accounted to 3252 images, in which 2484 are normal images, 744 are benign images and 600 are malignant images. The data augmentation process has been demonstrated in the Figure 13 where an image from original dataset is augmented into 12 different images. The data augmentation is performed using ImageDataGenerator function in Python Keras preprocessing library.

Figure 13

Example Image of Data Augmentation



Note. Example of Data Augmentation. Image on the left side is original image and right are the images augmented from the original image. Samples extracted from the mini-MIAS dataset (Suckling et al., 2015). Created using draw.io.

The breast mass plays a significant role in the identification and treatment of breast cancer as the information about the mass of breast helps in understanding of pattern of growth

and biological features. Usually, the breast masses that are classified as benign appear in regular shaped in the images and the masses with malignant classification tend to have margins that are irregular. Accuracy of segmentation of the breast masses is very crucial in classifying the types into benign or malignant. Segmentation is used to narrow down the information in the image and confine it to the Region of Interest (ROI) by dividing the image into multiple compartments to identify the masses. As the identification of masses is impacted by artifacts present in the image and information of pectoral muscles, such unwanted information is handled and removed in the preprocessing stage so that they will not confuse the model with irrelevant information. Some of the intangible benefits of segmentation are quantification of the tissue volume during analysis, identification of pathology, early detection, enhanced view of the structure of anatomy etc., (Michael et al., 2021).

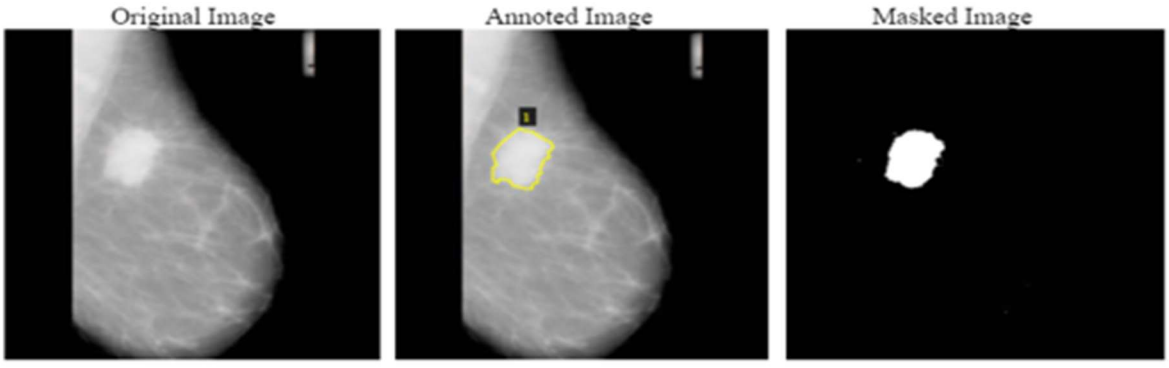
The different kinds of segmentation available are traditional, machine learning and deep learning segmentation. However traditional segmentation is accessible only to medical professionals with specific skillset and involves a tedious manual process. With the rise of mammographic image data that needs to be diagnosed, the traditional methods face the challenge of scaling up in terms of time and effort, as the medical experts need to identify the impacted region manually by comparing it to the rest of the information from the image. The machine learning segmentation methods require prior images that are truth evaluated by medical professional and need consistent preprocessing effort. The collaborative efforts between medical and machine learning experts poses a risk of some specific medical requirements that are ignored (Michael et al., 2021). The deep learning segmentation method outweighs both traditional and machine learning methods, as it can work directly on the image data without medical intervention and less preprocessing.

In the MIAS dataset that we are using for this project, the ground truth data is available but not the annotated images. Annotated masks are required on top of the truth data during training of the model to compare the annotated images with the masked images generated by the model.

VGG Image Annotator is an online software browser-based tool which is used to manually create annotations for the images. Selected image files are added to the Annotator project using Add Files tab. Then polygon shape is selected from the different region shapes that are available. Using the polygon shaped region select tool, identify, and draw the borders of the masses in the image manually. Selected annotations can be exported as JSON files to the desired storage path. Python code is used with OpenCV and NumPy libraries to produce masked images from the polygon coordinates data that is available from exported JSON files. The generated masks are stored in .png format (Talasila, 2021). The output of annotating and masking image is shown in Figure 14.

Figure 14

Generating Masked Images from Original Mammogram Images



Note. Generating Masked images from original mammogram images. The first image is original image, and second image is annotated image, and third image is masked image. Original image is extracted from the mini-MIAS dataset (Suckling et al., 2015). Created using draw.io.

The U-Net model is then fed with the original and masked images during training the model. The output of the U-Net model is the segmented map image as shown in Figure 15.

Figure 15

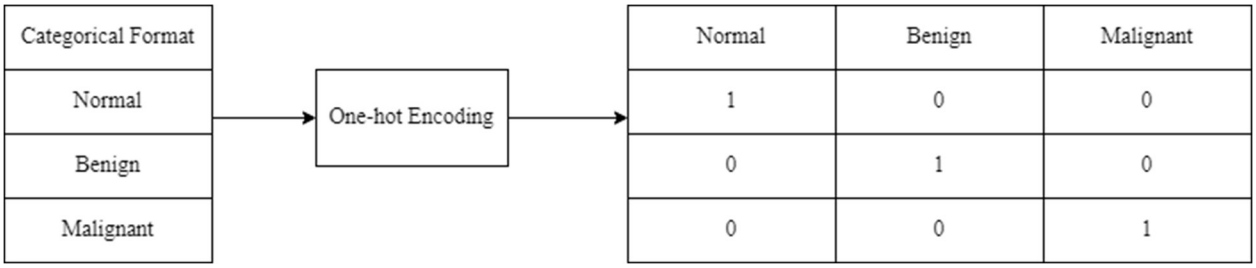
Input and Output for U-Net Model

Note. Input and Output for U-Net model Adapted from "Lesion Segmentation from Mammogram Images using a U-Net Deep Learning Network," by Neha S, 2020, *International Journal of Engineering Research & Technology (IJERT)*, *V9*(02), p. 4 (https://doi.org/10.17577/ijertv9is020213). Copyright 2020 by IJERT.

The label information on each mammographic image contains string format which is not acceptable by the model. Hence one-hot label encoding technique is used to convert the string format labels into numerical values. One-hot encoding technique is represented in tabular form with three columns namely normal, benign, and malignant as shown in Figure 16. If the identified image is malignant type, the malignant column is assigned with numerical value one and the rest of the columns were assigned values zero each. Similarly, the encoding table will be populated with corresponding information for each image based on the classification.

Figure 16

One hot encoding for Categorical Feature



Note. One hot encoding for Categorical Feature.

3.5 Data Preparation

A well-prepared data leads to effective analysis by providing the users with error free and accurate data. Data preparation involves formatting and enriching the data using standardizing formats to make it available for different analysis and prediction practices. A high quality and efficient data will lead to effective business decisions. The image data output from data transformation stage is stored in a folder. The new column by the name sample path is created for storing the path of the image file. By using Python code, the path of the corresponding image file is stored in the sample path column. There are a total of 3252 records in the dataset after transformation process. The resulting dataset after transformation phase is split into 80% and 20% ratio as training and testing datasets. The splitting of the dataset is achieved by using train_test_split function from sklearn.model_selection, which is a Python machine learning library. The dataset split is sometimes prone to data snooping where there is a statistical bias that is either voluntarily or involuntarily employed by the researcher. The data splits are properly divided based on the classes to ensure samples representing each of the classes were distributed and made available without any imbalances to the training and testing datasets. The training

dataset is again partitioned into train and validation sets using Cross Validation (CV) which is a re-sampling technique.

Since the dataset used in this project is not so large, CV technique is adopted which addresses the problem of improper distribution of data when some of the useful data points are excluded in the training dataset. Furthermore, in CV technique, K-Fold CV is used to distribute the datasets because this method is less biased over other cross validation techniques available (Manna, 2020). The distribution is achieved by the use of KFold function from sklearn.model_selection Python library. The K-Fold CV technique is based on a parameter k which denotes the number of folds that the dataset will be divided. Ideally the value of k falls anywhere between five and 10 (Manna, 2020). This project is based on 10-fold CV, which divides the 2601 records available from initial split between training dataset and testing dataset, to further classify into 2341 records to train and 260 records to test the data. This technique ensures that at least one-fold is available in the test dataset thereby preventing overfitting problem and remain effective even for new data.

During training, the model will be trained repeatedly on the training dataset for each epoch. During this training process the model will learn about the data features and predictions will be made based on what is learnt from the training. During training the model will classify the images into either normal, benign or malignant for each input in the training set. The loss is calculated for each input and weights are adjusted. And during the next epoch the model will continue to train on the same input again. The dataset that is allotted to training is used effectively to train all the three different models considered for detection. The three different models that operate on the training dataset are modified CNN, AlexNet, ResNet-50. The training dataset should be comprised of representation from every classification type of the images, so

that the neural network will be unbiased to any new samples for training. Sample of training dataset is shown in Figure 17.

Figure 17

Sample Data From Training Set

```
X_train.head(10)
```

	Ref_Num	ab_class	bg	Benign	Malignant	Normal	sample_path
258	mdb022_180_rotated	NORM	G	0	0	1	../samples\mdb022_180_rotated.png
232	mdb020_90_mirr_lr	NORM	G	0	0	1	../samples\mdb020_90_mirr_lr.png
33	mdb003_270_rotated	NORM	D	0	0	1	../samples\mdb003_270_rotated.png
157	mdb014_0_mirr_lr	NORM	G	0	0	1	../samples\mdb014_0_mirr_lr.png
148	mdb013_90_mirr_lr	MISC	G	1	0	0	../samples\mdb013_90_mirr_lr.png
93	mdb008_270_rotated	NORM	G	0	0	1	../samples\mdb008_270_rotated.png
37	mdb004_0_mirr_lr	NORM	D	0	0	1	../samples\mdb004_0_mirr_lr.png
139	mdb012_180_mirr_lr	CIRC	F	1	0	0	../samples\mdb012_180_mirr_lr.png
16	mdb002_90_mirr_lr	CIRC	G	1	0	0	../samples\mdb002_90_mirr_lr.png
217	mdb019_0_mirr_lr	CIRC	G	1	0	0	../samples\mdb019_0_mirr_lr.png

Note. Sample data from training set. Records extracted from the mini-MIAS dataset (Suckling et al., 2015).

The validation process is used to keep a check on the training stages at different epoch levels which is run at repeated intervals in the neural network and is responsible for tuning the hyperparameters in the network. The validation dataset is a separate set that is different from the training dataset and is used in the validation process. Validation on the data happens simultaneously while the training is in progress. Like training process, the validation process also classifies the data in each input based on what is learned during training process. The only difference between training and validation is that the weights will not be updated in the model based on loss calculated with validation set. Since the data in validation set is different from training set, the validation data does not consist of samples that the model is already aware with training. The main reason for having the validation set is to ensure that the data will not overfit the data in the training set. Sample of validation dataset is shown in Figure 18.

Figure 18

Sample Data From Validation Set

Ref_Num	ab_class	bg	Benign	Malignant	Normal	sample_path
mdb008_90_rotated	NORM	G	0	0	1	../samples\mdb008_90_rotated.png
mdb018_270_mirr_lr	NORM	G	0	0	1	../samples\mdb018_270_mirr_lr.png
mdb011_0_mirr_lr	NORM	F	0	0	1	../samples\mdb011_0_mirr_lr.png
mdb023_270_mirr_tp	CIRC	G	0	1	0	../samples\mdb023_270_mirr_tp.png
mdb002_180_mirr_tp	CIRC	G	1	0	0	../samples\mdb002_180_mirr_tp.png
mdb016_180_mirr_tp	NORM	G	0	0	1	../samples\mdb016_180_mirr_tp.png
mdb006_270_mirr_tp	NORM	F	0	0	1	../samples\mdb006_270_mirr_tp.png
mdb009_270_mirr_lr	NORM	F	0	0	1	../samples\mdb009_270_mirr_lr.png
mdb023_180_rotated	CIRC	G	0	1	0	../samples\mdb023_180_rotated.png
mdb009_180_rotated	NORM	F	0	0	1	../samples\mdb009_180_rotated.png

Note. Sample data from validation set. Records extracted from the mini-MIAS dataset (Suckling et al., 2015).

The test set is different from both training and validation sets. The major difference between training and validation sets and test set is that the data is not labeled in the test set. After the model is trained and validated, then the test data is used for predictions. Sample of testing dataset is shown in Figure 19.

Figure 19

Sample Data From Test Set

Ref_Num	bg	sample_path
mdb003_180_rotated	D	../samples\mdb003_180_rotated.png
mdb011_90_mirr_lr	F	../samples\mdb011_90_mirr_lr.png
mdb017_90_mirr_lr	G	../samples\mdb017_90_mirr_lr.png
mdb011_180_mirr_lr	F	../samples\mdb011_180_mirr_lr.png
mdb019_0_rotated	G	../samples\mdb019_0_rotated.png
mdb020_180_rotated	G	../samples\mdb020_180_rotated.png
mdb013_0_rotated	G	../samples\mdb013_0_rotated.png
mdb017_270_mirr_tp	G	../samples\mdb017_270_mirr_tp.png
mdb022_0_mirr_tp	G	../samples\mdb022_0_mirr_tp.png
mdb012_270_mirr_tp	F	../samples\mdb012_270_mirr_tp.png
mdb007_180_mirr_lr	G	../samples\mdb007_180_mirr_lr.png

Note. Sample data from test set. Records extracted from the mini-MIAS dataset (Suckling et al., 2015).

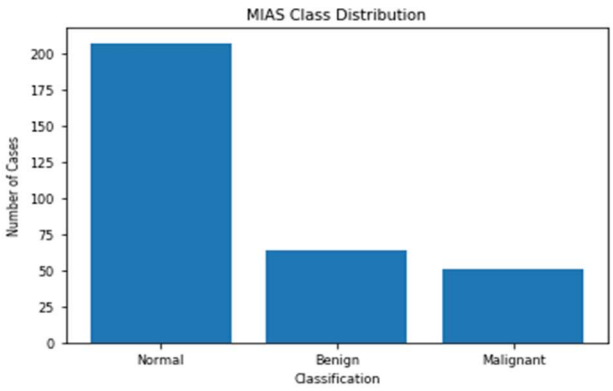
3.6 Data Statistics

Any machine learning project is greatly dependent on the data statistics to interpret and evaluate the output. Statistics in simple terms means the use of different mathematical models to perform analysis on the data. Data Statistics is important to organize the data at different stages of the project which includes getting rid of the information that is not required and cataloging the data which is necessary in an efficient and effortless manner (Rawat, n.d.). Data Statistics play a major role in making predictions and classifying the data to derive useful insights for the clients in the form of reports, charts, or dashboards (Rawat, n.d.). Detection of patterns and grouping the data by selecting the optimal data and removing the unnecessary data points is also part of data statistics.

The raw dataset used in this project is considered from MIAS mini database which consists of 322 mammogram images altogether out of which 208 are classified as normal, 63 as benign and 51 as malignant as shown in Figure 20.

Figure 20

Bar Chart for MIAS Class Distribution

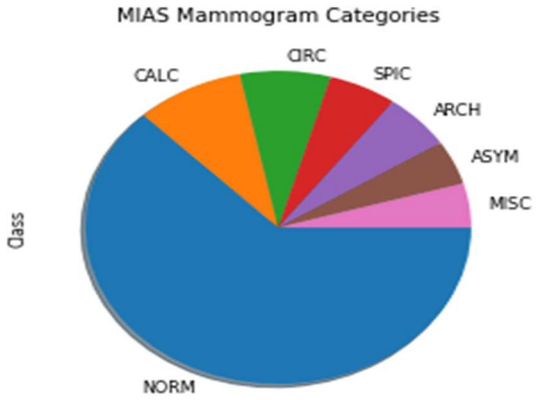


Note. Bar chart for MIAS class distribution.

From Figure 20 we can see that this data set is vigorously imbalanced as the appropriation is not uniform. The ratio of data for normal cases to benign and malignant cases is disproportionate. It is essential that we avoid data imbalance before feeding it to the model to avoid bias.

Figure 21

Pie Chart for MIAS Mammogram Categories

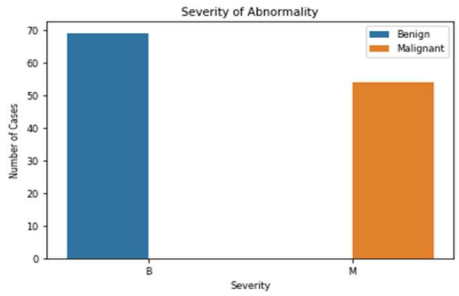


Note. Pie Chart for MIAS Mammogram Categories.

The abnormalities are classified into seven categories which are Calcification, Circumscribed masses, Architectural distortion, Spiculated masses, Other ill-defined masses, Asymmetry masses and Normal masses as shown in the Figure 21.

Figure 22

Bar Chart for Severity of Abnormality



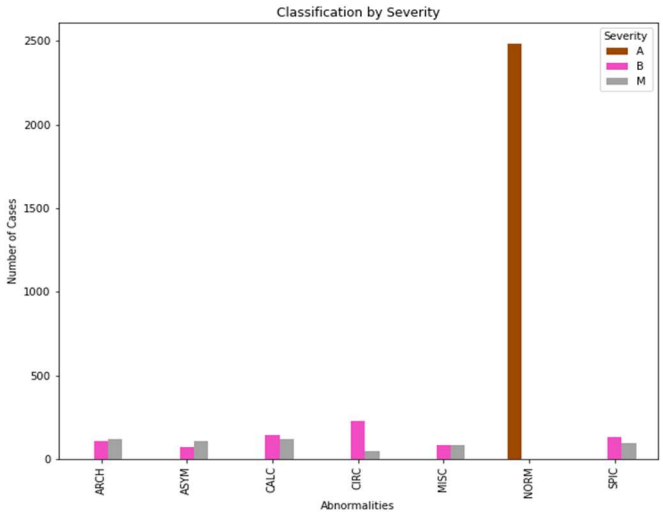
Note. Bar Chart for Severity of Abnormality.

Figure 22 is used to represent the severity of abnormalities found from the collected raw samples.

During the cleaning process in preprocessing stage, the images with missing information are identified and removed. Six classes of abnormalities and count of each are illustrated in Figure 23 and Figure 24. The consolidated list of images after preprocessing accounts for a total of 319 images in which 207 are normal images, 62 are benign images and 50 are malignant images. Since the dataset is too small for the model, the data augmentation process is employed to augment the data which resulted in generating 12 images for each of the image in raw dataset. The total images after data augmentation process are 3828 images out of which 2484 are classified under normal, 744 benign and 600 are malignant.

Figure 23

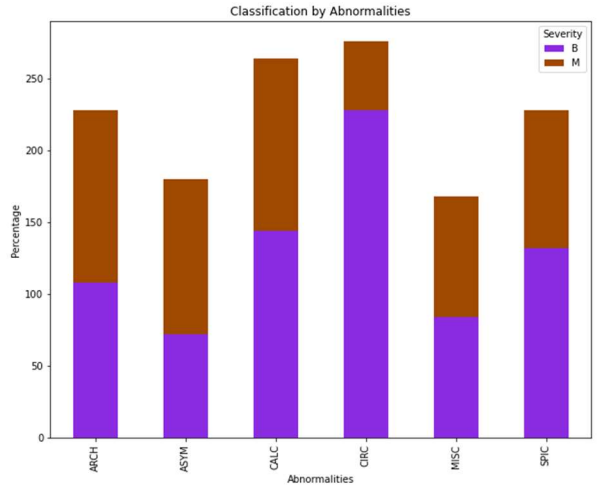
Classification of Severity After Data Augmentation



Note. Classification of severity after data augmentation.

Figure 24

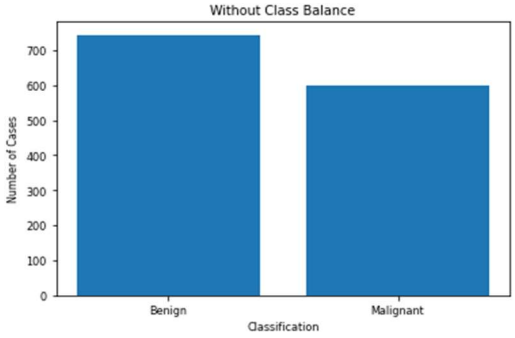
Classification of Only Abnormalities After Data Augmentation



Note. Classification of only abnormalities after data augmentation.

Figure 25

Benign and Malignant Cases Before Balancing



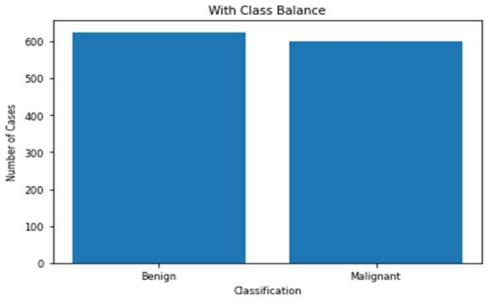
Note. Benign and Malignant cases before balancing.

Since the data imbalances are observed from the Figure 25, under sampling method is used to balance the classification of data as shown in Figure 26 and Figure 27. There is no statistical change after data segmentation phase because segmentation is focused on enhancement of images rather than change in the quantity of the image samples. After data

segmentation, new columns are introduced to the existing metadata table due to the use of sample path for storage location and one-hot label encoding technique to classify the image labels. The total columns on the metadata table after image segmentation changed to seven and the total images remained at 3828 number.

Figure 26

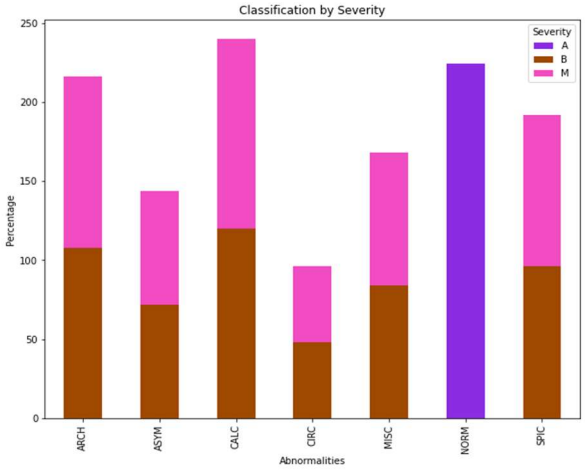
Benign and Malignant Cases After Balancing



Note. Benign and Malignant cases after balancing.

Figure 27

Classification of Severity After Data Balance



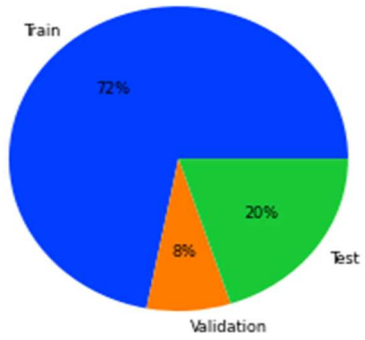
Note. Classification of severity after data balance.

In the data preparation phase, the dataset is parted into 80% for training and 20% for testing which resulted in 3062 mammogram images for training and 766 mammogram images

for testing. The training dataset is segregated into train and validation datasets based on 10-fold cross validation technique which resulted in 2755 mammogram images for training and 307 mammogram images for validation as shown in Figure 28.

Figure 28

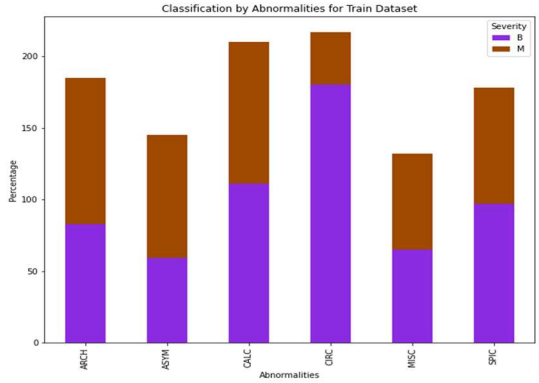
Pie Chart Representing the Percentage Split of Data into Train, Test, and Validation Sets



Note. Pie chart representing the percentage split of data into train, test, and validation sets

Figure 29

Classification of Abnormalities After Data Split for Training Data

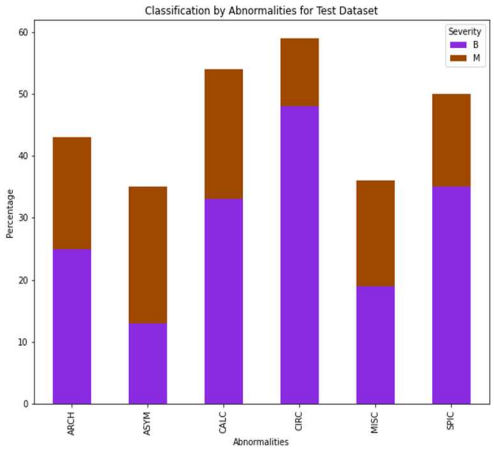


Note. Classification of abnormalities after data split for train data.

Figure 29 is used to represent the classification of abnormalities which can be either benign or malignant in the training dataset.

Figure 30

Classification of Abnormalities After Data Split for Testing Data



Note. Classification of abnormalities after data split for test data.

Figure 30 is used to represent the classification of abnormalities which can be either benign or malignant in the testing dataset.

4. Model Development

4.1 Model Proposals

Diagnosis of symptoms that lead to breast cancer during initial stages has proven critical in combating the disease. Different screening approaches are used to diagnose breast cancer, and mammography is one of the most extensively used and cost-effective screening procedures (Michael et al., 2021). Mammography is primarily based on capturing the images of breast tumors and their analysis based on the captured image properties. However incorrect interpretations and errors occurring during the translation of mammographic image data have led to improper decisions made by the medical professionals and eventually reducing the chances of survival for the affected patients.

In order to reduce the human interpretation errors involved, CAD has come into picture and seems to deliver promising results. CAD systems are broadly classified as traditional and deep learning systems. However, traditional CAD systems still require the intervention of medical professional or a clinician while identification of image features of impacted regions of the breast. Deep learning CAD systems like CNN proved to overcome the challenges in traditional systems by learning the features of the mammography images through different layers. The CNN technique has scored more accuracy than the machine learning algorithms and has become an effective alternative in the assistance of diagnostics for breast cancer detection (Jimenez-Gaona et al., 2020).

This project analyses CNN techniques with different training approaches to identify the best detection model that is more accurate and less time consuming. The different models adopted in this project are modified CNN trained from scratch, AlexNet and ResNet-50. In training from scratch approach, the model is trained with the available mammogram images from

the mini–MIAS dataset whereas in AlexNet and ResNet-50 rely on pre-trained model called transfer learning. The use of segmentation information from U-Net is the key aspect in contributing to the efficacy of CNN, where the performance is improved in the diagnosis. The use of transfer learning in AlexNet and ResNet-50 reduces the computing time and cost which are important factors in determining the right model. The reason for considering CNN models is that the model outranks the predecessor and similar models in terms of better and accurate feature extraction, adaptability to run on different devices and better computational time.

There are different layers that are employed in building the CNN model. Convolution layers, max pooling layers and fully connected layers constitute the architecture of CNN. Mammogram image input data is fed to the convolution layer. Convolution layer is based on transformation of input images into feature map by application of filters of specified matrix size. Filters identify patterns in images such as multiple edges, forms, textures, etc., and each convolution layer will have a set of different filters. The purpose of the filter is to convolve on each block of specified pixels received as input to the convolution layer. The filter iterates through the input over the specified block of pixels from which the dot product of the pixels is computed and stored. The filter will proceed with next set of blocks to repeat this process of computation and storage of the result. Once this filter has been slid across the entire input, a new representation of the input is constructed using the filter's stored dot products. The output of the convolution layer is this matrix of dot products. The output generated as feature maps is provided as transformed input to the next layer and this operation is termed as convolution operation. The feature map size of the generated output is calculated using the Equation 1, where n denotes input features count, m denotes output features count, k represents size of kernel, s is stride size and p is padding size of Convolution (Hien, 2018).

$$m = \left[\frac{n + 2p - k}{s}\right] + 1 \quad (1)$$

Figure 31

Pseudocode of convolution layer

Algorithm 1 Pseudo code for a convolutional layer

```
for i from 1 to m do                    —inter-output
    for j from 1 to n do                —intra-output
        for r from 1 to R_o do
            for c from 1 to R_o do
                tmp = 0
                for ii from 1 to k do
                    for jj from 1 to k do
                        tmp = tmp + K[ii][jj] × X[j][s × (r − 1) + ii][s × (c − 1) + jj]
                    end for
                end for
                Y[i][r][c] = Y[i][r][c] + tmp
                if j == n
                    Y[i][r][c] = f(Y[i][r][c] + bias)
                end if
            end for
        end for
    end for
end for
```

Note. Pseudocode of convolution layer. Adapted from "Automatic Code Generation of Convolutional Neural Networks in FPGA Implementation.," by Liu et al., 2016, *International Conference on Field-Programmable Technology (FPT),* p.2. (https://doi.org/0.1109/FPT.2016.7929190). Copyright 2016 by FPT.

Figure 31 represents contains pseudo code used in convolution layer and it can be observed that for each pixel in the output corresponding bias value is added. Function f is used to restrict the range of pixels to a desired value (Liu et al., 2016).

Pooling layer is used to complement the model with spatial variance, which means the ability of the system to identify the pattern in an image with different changes in appearance (Arc, 2018). Max pooling and average pooling techniques are used across different models in this project. By lowering the number of pixels in the convolution layers output, max pooling minimizes the size of feature maps. To perform max pooling, we must first determine the size of

the filter and the stride by which the filter should move as it slides across the image for example, if we use a 2x2 filter with stride two, the first 2x2 region from the convolution output is taken and the max value from that block is calculated and stored as shown in the Figure 32, then the filter is moved by the number of pixels defined in the stride to calculate max value (Arc, 2018). This process is carried out through the entire input and finally the output is generated with dimensions reduced by a factor of two. Since max pooling reduces the resolution of a convolution layer's supplied output, the network will have high-valued pixels in the future, which minimizes the number of variables in the network and thus reduces computational load (Arc, 2018). Like max pooling, average pooling is aimed at calculation of average value of the specified block.

Figure 32

Max Pooling Operation on a Single Slice of Input Image

Note. Max Pooling Operation on a Single Slice of Input Image. Adapted from "*Convolutional Neural Network - Towards Data Science*.," by Arc, 2018, *Medium*, p.5. (https://towardsdatascience.com/convolutional-neural-network-17fb77e76c05), Copyright 2018 by Medium.

Fully connected layers or dense layers, like artificial neural networks, are a dense network of neurons. In convolution and pooling layers, the features of the input image are

extracted. To categorize the image to a certain category, the corresponding features are learned using fully connected layers. The output from either layers of the convolutional or max pooling is reduced to a one-dimensional array which is then served as input to the dense layers and there is connectivity between neurons across different dense layers (Arc, 2018). The output categories are correlated to the neutron count that are present in the output layer, and the categorization of images is achieved with the use of softmax activation function. There are three output categories in this project which are benign, malignant, and normal. So, the final fully connected layer will have three neurons. In fully connected layer, the output vector is calculated using Equation 2 where vector length is represented by n whereas X refers to the input and W refers to the weights (Liu et al., 2016).

$$Y_i = \sum_{j=1}^{n} X_j * W_{ij} \tag{2}$$

Figure 33

Pseudo Code of Fully Connected Layer

Algorithm 2 Pseudo code for a fully-connected layer

```
1: for i from 1 to m do
2:     tmp = 0
3:     for j from 1 to n do
4:         tmp = tmp + W[i][j] × X[j]
5:     end for
6:     Y[i] = tmp
7: end for
```

Note. Pseudocode of fully connected layer. Adapted from "Automatic Code Generation of Convolutional Neural Networks in FPGA Implementation.," by Liu et al., 2018, *International Conference on Field-Programmable Technology (FPT),* p.2 (https://doi.org/0.1109/FPT.2016.7929190). Copyright 2018 by FPT.

In Figure 33 different groups of inner-product vectors are represented by m and the vector length is represented by n whereas X refers to the input and W refers to the weights (Liu et al., 2016).

Activation function is a key element in the construction of a neural network. The working of the activation function is in close resemblance to the biological process of the brain function where different stimuli fires different neurons. Activation function determines which neuron to be activated and which neuron to be not activated based on the input that is relevant to the model prediction, as it is connected with every neuron in the network. The sigmoid function, often known as the squashing function, is a form of activation function. If the input to the sigmoid function is a negative number, it will be transformed to a number close to zero and if the input is a positive number, it will be transformed to a number close to one. However, for very high and very low input values, there will be no change in the prediction which results in a vanishing gradient problem. ReLU activation function that turns the outcome to zero if the supplied value is less than or equal to zero, and if value is higher than zero output carries the same value as input. Because ReLU is computationally efficient, the network can converge quickly. Mathematically ReLU activation function is represented in Equation 3 where y is output, and x is input. The gradient of the function becomes zero when the input approaches zero or is negative and the network cannot complete back propagation, which is known as dying ReLU. To address the dying ReLU problem, a leaky ReLU was devised, which replaces a small negative slope of 0.01 when the input is less than zero. However, with negative input values leaky ReLU does not produce reliable predictions. SoftMax is exponential, pushing one result near to one while pushing another close to zero. It converts scores into probabilities and the cross-entropy cost function is frequently computed for SoftMax output. Mathematically softmax activation function

is represented as shown in the Equation 4 where σ is softmax, K is number of classes, e^{z_i} and e^{z_j} are exponential functions for input and output vector (Gupta, 2020).

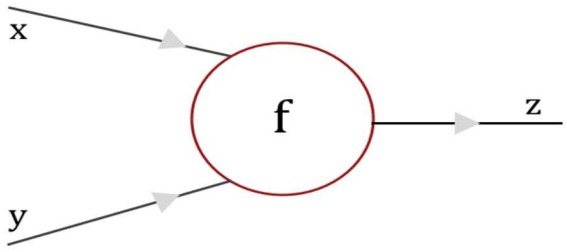
$$y = max(0, x) \tag{3}$$

$$\sigma(\vec{Z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{4}$$

Every time we transmit input through the model, it passes on through the network layers to the output layer. The input for each node in the existing layer is derived from the output of the preceding layer and this input is calculated as the combined sum of weights at every connection multiplied by previous layer's output. The output at a particular node is formed by passing the weighted sum through the activation function. The output this derived is then passed further as input to the subsequent node and this process continues further until the output layer is reached. This process is referred as forward propagation and the function f is represented in the Figure 34 where x and y are input variables, z is output calculated from the function (Solai, 2020).

Figure 34

Example for Forward Pass

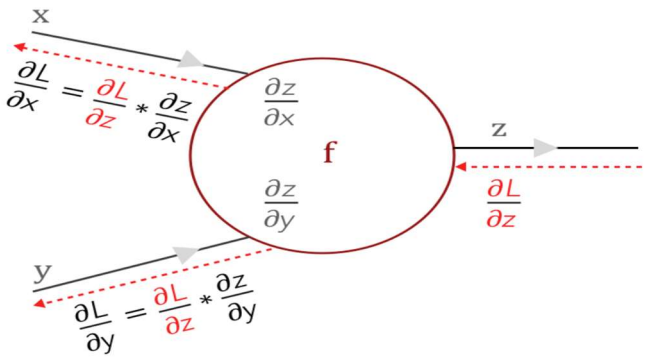


Note. Example for Forward Pass. Adapted from "*How does Backpropagation work in a CNN?*" by Solai, 2020, *Medium*, p.7. (https://pavisj.medium.com/convolutions-and-backpropagations-46026a8f5d2c), Copyright 2020 by Medium.

The backpropagation algorithm is a prominent feedforward method used in neural network training. During the training of CNN model, the back propagation process is used by the optimizer. The weights at each epoch are updated by the optimizer to minimize the loss. The loss function derivate is calculated based on the weights in the model. The update of weights is done with the gradient or from the loss function derivative which were calculated. Gradient descent updates the weights by using activation outputs from output nodes to increase or decrease the weights based on the loss. For each sample, Stochastic Grade Descent (SGD) adjusts the output values for correct and incorrect nodes, thus lowering the loss. The output value is increased for the correct node and decreases for the incorrect node. This process is iterated till the input layer is reached. This process is referred as backward propagation and the function is represented in Figure 35 where $\frac{\partial z}{\partial X}$ and $\frac{\partial z}{\partial y}$ are local gradients and $\frac{\partial L}{\partial z}$ is the loss previous layer (Solai, 2020).

Figure 35

Example for Backward Pass



Note. Example for Backward Pass. Adapted from "*How does Backpropagation work in a CNN?*" by Solai, 2020, *Medium*, p.8. (https://pavisj.medium.com/convolutions-and-backpropagations-46026a8f5d2c), Copyright 2020 by Medium.

The primary goal of an optimizer during training is to reduce the difference between actual and anticipated outputs in a given set of samples. Weights are set arbitrarily during

training and then incrementally updated as it gets closer to minimizing the loss. The size of these steps is dependent on the learning rate. Following the calculation of loss for the supplied inputs, the gradients of loss for each of the weights in that model are computed. The multiplication factor is derived for computed values and the learning rate. Learning rate is a small integer that ranges from 0.01 to 0.0001 and is a hyper parameter that must be tested and tuned with each model before a value is selected. The value of learning rate should not be either very large or very small. There is a risk of overshooting when a learning rate number is set on the upper side. To avoid the risk of overshooting, if the number for learning rate is set on the lower side, the step size would be very small and would take more time to achieve the point of minimum loss. When updating weights, adding a velocity term aid in accumulating momentum towards minima with each update, which is known as the momentum algorithm. The weight of previous updates in the weighted average is determined by the value of momentum. Adam optimizer is a hybrid of RMS prop and momentum optimizer that employs adaptive learning rates. (Zhang et al., 2020).

The distance between anticipated and true output values is calculated using the loss function and it is achieved using conditional maximum likelihood estimation. Weights and bias terms are chosen to maximize the log probability of true valued instances in the training set while conducting conditional likelihood estimation. When the log probability is maximized, it gives a negative log likelihood loss, also known as Cross Entropy loss. The difference between the probability distribution of expected and actual values is measured by Cross Entropy. The Cross Entropy formula is very easy to calculate, and it is the sum of actual probabilities multiplied by the log of expected probability over all classes in the distribution. The degree of Cross Entropy loss varies by the distance between actual and predicted values. When the distance is large between true and predicted values there is a high Cross Entropy loss and when the distance is

small, Cross Entropy loss is low. Categorical Cross Entropy to compute loss in the models of this project, which is represented in Equation 5, where actual label is represented as t_i and p_i is probability obtained from softmax function and n is number of classes (Koech, 2021).

$$L_{CE} = -\sum_{i=1}^{n} t_i \, log(p_i) \tag{5}$$

The deep learning segmentation method outweighs both traditional and machine learning methods, as it can work directly on the image data without medical intervention and less preprocessing. There are different deep learning segmentation techniques available like U-Net, RU-Net, FCN, SegNet, etc. Although the convolutional network has positive outcomes in image detection tasks, the outcomes are criticized for success rates that are lower (Michael et al., 2021). Fully Convolutional Network (FCN) has addressed the challenges of CNN with dense prediction algorithm that is independent of fully connected layers which can result in creating the image segmentation map in less period (Neha S, 2020). Furthermore U-Net is an extension of FCN with a U-shaped architectural pattern. U-Net is widely used deep learning model due to the fact that it does not need more annotated images and can handle networks with multiple layers with its high GPU computing performance (Neha S, 2020).

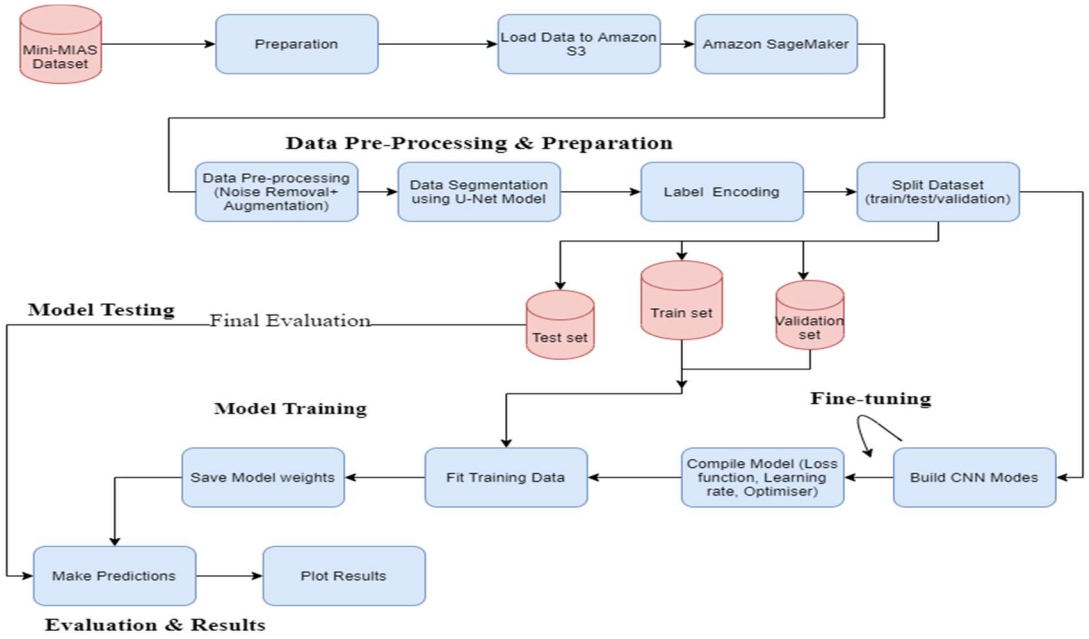
As part of data transformation U-Net model is used to generate predicted segmented map images. The data from the original dataset along with masked image data is classified into three sections to be fed to training, testing and validation. From the original data 70% is allotted for training, 20% for testing and 10% to validate the model.

4.2 Model Supports

The data considered for analysis in this project is downloaded from mini-MIAS database into the local machine. Using the Command Line Interface specific to Amazon AWS, the dataset from local machine is synchronized with Amazon S3 Cloud Storage. Jupyter Notebook is the software platform that is based out of Python and it is created in Amazon Sagemaker IDE on ml.p3.2xlarge instance in the AWS Cloud to serve as the basis of development. The data in AWS S3 storage is loaded into Jupyter Notebook in Amazon Sagemaker by using Boto3 library in Python. The next step in the data flow consists of data preprocessing where Python Imaging Library (PIL), cv2, NumPy, json, matplotlib, seaborn, pandas and Keras libraries are used. U-Net model is used in data transformation stage where the image data is transformed into segmented maps. VGG Image Annotator tool is used in creating annotations for the mammogram images during transformation. The train_test_split function from the Scikit-learn module is used to split the segmented data. TensorFlow library is used to implement the model which has Keras modules embedded. The functions in Keras modules like models, layers, optimizers, losses, and callbacks are used in the development of the model. For model evaluation and model validation metrics and KFold functions are used respectively from sklearn library. All the above-mentioned libraries were loaded into the Jupyter Notebook to be used in different stages of project implementation. Figure 36 represents the data flow and the different components involved in various stages of implementation.

Figure 36

Detailed Data Flow Representation



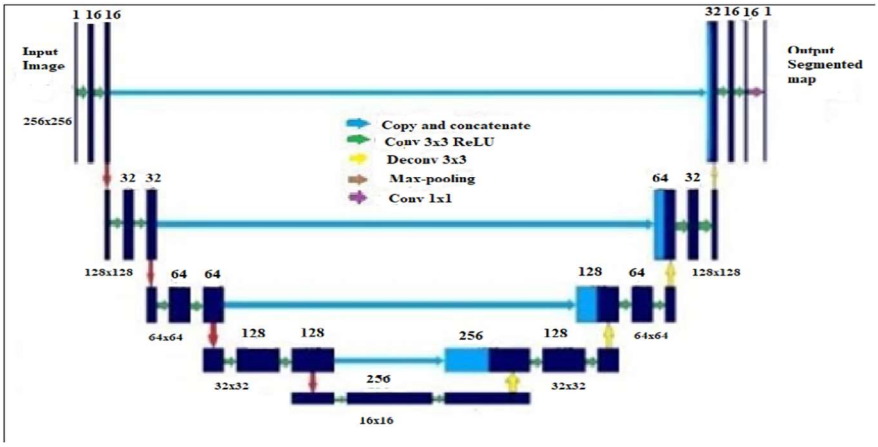
Note. Detailed Data Flow Representation of this project.

Semantic segmentation is based on the aspect that every pixel in the images is painted either to represent a mass or a background. The U-Net architecture comprises of orchestrating different deep learning tools like convolution, max pooling and dense layers which will take image as input and produce segmentation mapped image as output. The architecture in U-Net model consists of two paths as shown in Figure 37. The left path is called contraction or encoder path and the path on right is called expansion or decoder path. The U-shaped connection between contraction and expansion paths connects the high-resolution image features from encoder with features that were up sampled from decoder. The contraction path consists of two 3x3 convolution layers along with an input layer and the depth of these layers is 256x256x1. The number one denotes the number of channels, which is gray scale in our case. The pixels in each axis are represented by 256x256. The next layer is max-pooling layer with stride two 2x2 matrix.

The max-pooling layer selects the maximum value in the matrix and replace the matrix with that value. Dimensions of the image in this layer are dependent upon the type of padding. Here the padding type is selected as same, which adds extra pixels to image edges so that the output image is same as the input image. At every stage of down sampling the feature channels are doubled since we have considered stride two. The decoder path consists of a deconvolutional layer of 2x2 matrix and two 3x3 convolution layers without padding. The up sampling of feature map takes place in the expansion path. The feature maps derived from both shallow and deep layers are concatenated by the skip connections due to which we get localized information that makes semantic segmentation possible. The last convolution layer which is 1x1 matrix is used obtain the required classification result. Adam optimizer, Binary Cross Entropy is used (Neha S, 2020).

Figure 37

U-Net Architecture



Note. U-Net architecture. Adapted from "Lesion Segmentation from Mammogram Images using a U-Net Deep Learning Network," by Neha S, 2020, *International Journal of Engineering Research & Technology (IJERT)*, *V9*(02), p.3 (https://doi.org/10.17577/ijertv9is020213). Copyright 2020 by IJERT.

The focus of this project is aimed at working on the determination of breast cancer growth and to accomplish that, three CNN models were considered. In most of the existing approaches, the raw image data is provided as input to CNN models to detect breast cancer. But the existing processes involve significant amount of computational time and operational costs. The proposed model is targeted to achieve high accuracy with reduced computational time and operational costs. In the current approach of CNN, the convolution layer is provided with the dataset of images in the form of segmentation maps.

The proposed CNN architecture has four average pooling layers, six convolutional layers, and three fully connected layers. Every convolution layer is activated by ReLU. The input data is initially provided as input to the convolution layer with four filters each of size 3 x 3 with padding as three. First layer output is again provided to the subsequent convolution layer with four filters of 3 x 3 size with padding as three. The next layer is the average pooling layer with 2 x 2 filter and stride two. The next layer is a convolution layer with 3 x 3 filter and a total of 16 filters with padding as two. Following the convolution layer, is an average pooling layer with the same specification as all other pooling layers have. The next layer is again a convolution layer with filter size, filter number and padding information as the previous layer. Followed by convolution is again an average pooling layer with same specifics as previous pooling layers. There are two subsequent convolution layers after the average pooling layer, with number of filters as 80, filter size of 3 x 3 and padding as one. The succeeding layer is again an average pooling layer of 2 x 2 filter and stride one. The output from the average pooling layer is flattened and fed to dense layers (Charan et al., 2018).

There is a softmax activation function that is applied on the last fully connected layer which has three neurons that signify three classifications. The loss is computed using Categorical

Cross Entropy and Adam is used as an optimizer. The architecture diagram for the proposed CNN model is represented in the Figure 38.

Figure 38

CNN Architecture

Note. Proposed CNN Architecture Adapted from "Breast Cancer Detection in Mammograms Using Convolutional Neural Network," by Charan, S., Khan, M. J., & Khurshid, K. 2018, *International Conference on Computing, Mathematics and Engineering Technologies* (ICOMET), p.3 (https://doi.org/10.1109/ICOMET.2018.8346384). Copyright 2018 by ICOMET.

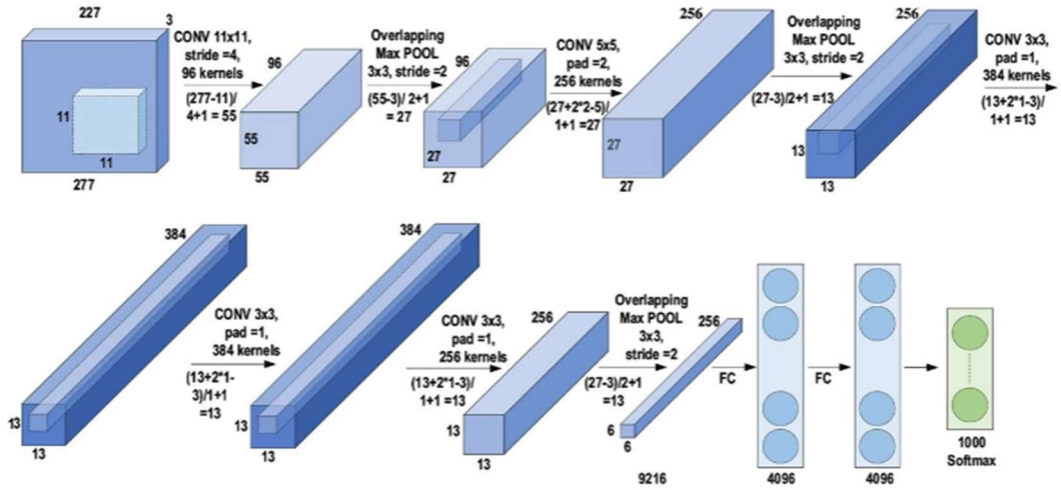
AlexNet is another CNN model that is considered in this project. The architecture of AlexNet is comprised of different layers that involve five layers of convolution, three layers of max pooling, two layers of fully connected and a softmax layer. Input to AlexNet is an RGB image which has three channels with 227x227 dimensions. AlexNet contains more than 60 million parameters, and it has around six lakhs of neurons (Wei, 2020). The AlexNet network is split into two pipelines which makes it possible to train the model with two GPUs. ReLu is the activation function used in AlexNet, AlexNet uses momentum optimizer technique as it is focused on gradient with momentum optimizer (Wei, 2020).

First convolution layer has kernel of size 11x11 with stride four and it has 96 different Kernels. The output of the first convolution layer contains 96 different channels and each of this feature map is of size 55x55. Next is overlapping max pooling layer where max pooling operation is done with a window size of 3x3 and stride two. After performing max pooling operation, every feature map is adjusted from 55x55 to 27x27, and the number of channels corresponding to the feature is 96. The second is a convolutional layer in AlexNet of 256 filters of size 5x5 with padding as two. Padding is used to keep the output feature size the same as the input feature size. The number of output channels from this layer is equal to the number of kernels which is 256, with every feature map of size 27x27. An overlapping max pooling layer with 3x3 window size and stride as two succeeds convolution layer. After performing max pooling operation, the output from the max pooling layer has 256 channels with every feature map of size 13x13. Max pooling layer is followed by three consecutive convolution layers in which the first layer consists of 384 filters with 3x3 size and padding is equal to one. The output of convolution layer is passed to the second consecutive convolution layer of size 3x3 and padding is equal to one. There are 384 feature maps at this point and every feature map is of size 13x13. There are 256 kernels in the third consecutive convolution layer. Max pooling operation is performed with a window size of 3x3 and stride equal to two, which gives the output of 256 feature maps with 6x6 size in the next layer. There are two dense layers and a final output SoftMax layer after max pooling operation (Alzubaidi et al., 2021).

The original AlexNet has a SoftMax layer with 1000 output channels which is replaced with three output channels for this research to categorize whether the tumor is normal, benign, or malignant. Architecture of AlexNet is shown in Figure 39.

Figure 39

Architecture of AlexNet

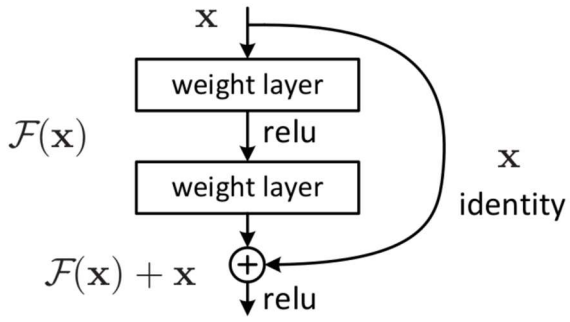


Note. Architecture of AlexNet Adapted from "Review of Deep Learning: Concepts, CNN Architectures, Challenges, Applications, Future Directions" by Alzubaidi et al., 2021, *Journal of Big Data,* p.28 (https://doi.org/10.1186/s40537-021-00444-8). Copyright 2021 by Journal of Big Data.

ResNet is a residual neural network with individual residual blocks in it. At each block of convolutions, there are sets of feature maps that are constantly being downsized. At some point these feature maps will be small and cannot be downsized anymore. The dimensionality can be maintained by adding convolution layers and performing convolutions. For a sufficiently deep model, adding another layer can introduce a degradation problem. Deep network model can calculate a strong set of features for its task at a certain depth. The next block should aim to be a copy of the previous block which is referred to as identity mapping. Degradation results suggest that there is difficulty in learning this identity mapping. Microsoft came up with a solution for the degradation problem. Instead of trying to learn the identity function, pass the information forward with a skip connection which is also called residual learning (Kaushik, 2020).

Figure 40

Residual Network



Note. Residual Network Adapted from "*Understanding ResNet50 architecture*" by Kaushik, A, 2020, *OpenGenus IQ: Computing Expertise & Legacy,* p.5. (https://iq.opengenus.org/resnet50-architecture). Copyright 2020 by OpenGenus IQ.

From the Figure 40 we can see that x is an identity feature and the residual mapping is denoted in Equation 6 and is also called a feedforward network with skip connections. In order to update the weights during training, error gradients can be passed between any layers through the shortcut connections (Kaushik, 2020).

$$H(x) = F(x) + x \tag{6}$$

ResNet-50 architecture consists of different layers that sums up to 50 layers. There are 49 convolution layers with the combination of different kernels sizes with varying numbers and one dense layer. Convolution layer is the first layer in the architecture with 7x7 kernel and 64 kernels with stride size of two. A max pooling layer of 3x3 size and stride two is followed by the convolution layer. The next set of layers are a total of nine convolution layers with sizes 1x1, 3x3 and 1x1 with number of kernels as 64, 64 and 256 respectively. Following the nine convolution layers, is a set of 12 convolution layers with kernel sizes 1x1, 3x3 and 1x1 with number of kernels 128, 128 and 512 respectively. Subsequently 18 layers of convolution of sizes

1x1, 3x3 and 1x1 with number of kernels 256, 256 and 1024 succeeds the 12 layered convolutions in the previous layer. The 18 layered convolutions are followed by a set of nine convolution layers of sizes 1x1, 3x3 and 1x1 with number of kernels 512, 512 and 2048. Average pooling operation is performed after the above convolution layers and the output from the average pooling is flattened and provided as an input to dense layers. The final layer is a dense layer with three nodes that ends with a softmax function. The pooling layers and activation functions are not counted as part of the layered architecture in ResNet-50 (Kaushik, 2020).

4.3 Model Comparison and Justification

The increasing number of breast cancer cases have made it difficult for the medical diagnostic professionals to cope up with the traditional approaches in the detection of breast cancer. Although automated analysis of mammographic images with the use of different computational mathematical models using machine learning algorithms is in place since past few decades, the feature extraction of impacted region from the mammogram images still requires manual intervention to some extent for traditional machine learning algorithms. Besides high accuracy and effectiveness, the machine learning algorithms have certain limitations as they cannot operate directly on raw images. They required significant effort for preprocessing and feature extraction from raw image data. Feature extraction by learning from the features of optimized and filtered data from raw images have laid the foundation for deep learning methodologies (Litjens et al., 2017).

CNN is a kind of deep learning model that has proven to be effective in feature extraction and image analysis. CNN model is complemented by the advances in computational resources like powerful GPUs and scalable processing instances like Amazon Sagemaker, thus making it very efficient in crunching interpretation data from very high-quality images at a faster rate. The

self-reliant learning patterns, efficient processing abilities and adaptability with different systems have made CNN outperform the existing models in the detection of breast cancer (Litjens et al., 2017). The most important feature of CNN is automatic detection of important features from image dataset without any human intervention. CNN model developed from scratch involves convolution, pooling, padding and full connected layers.

Convolution layers aim at feature extraction using filters and refining the extracted features for generalized representation using pooling. Padding is used to improve the edge features and maintain standardized sizes for the output. Fully connected layer finally categorizes the extracted features to form a generalized output. The downside of CNN model is that it involves high computational cost and requires huge amount of data to train the model as summarized in Table 2, without which it might run into problems like overfitting, underfitting etc. Also, if the CNN model is not equipped with the suitable GPU, computational delays in training the model occur. To supplement the CNN model with large amount of data, transfer learning technique is adopted (Sarkar, 2018).

The use of pre-trained network is always preferable, because the number of parameters that must be trained rises as the depth of a neural network increases. Transfer learning finds its importance particularly in use cases related to computer vision, where the features that are learned from large image sets like ImageNet are highly reusable for a different variety of tasks based on image recognition. For example, a model that has been trained on categorizing objects can be fine-tuned to support another use case where different scenes should be classified. There are several ways to transfer knowledge from one model to another. One method for transferring information from one model to another is to replace the output layer of an already trained model with a randomly initialized layer and letting the parameters remain fixed in all other layers while

training the parameters only in the top layer for the new job. This method works best if the new model and the previously trained model have similarities with respect to data and tasks. Transferring the initial values of parameters and initializing weights by utilizing a pre-trained model rather than initializing randomly might give the model a strong start and speed up convergence if there is sufficient data available to train a model from scratch (Sarkar, 2018).

When a CNN is trained on a huge set of images, the filters in the initial layers are more inclined towards learning of generic features such as patterns, edges, textures etc., rather than the features specific to a particular classification task. For example, in this context, the images are classified based on the abnormality such as benign, malignant, and normal. In the initial layers this level of classification is not achieved from the filters. As we move closer towards the output, the layers tend to learn more features specific to the required classification task. For example, if the final layer in the network is trained with a view to perform only certain classification tasks, that layer will be confined to that classification task.

AlexNet is one of the CNN models that is considered for analysis in the detection of breast cancer. It is a pre-trained model that is based on transfer learning. AlexNet is considered faster than the above CNN model that is trained from scratch as it is the first model which is based on GPU and uses ReLU activation function. AlexNet was developed to achieve better accuracy at a faster rate as the training speed is greatly improved in this model when compared to CNN model trained from scratch. The improvement of training speed is achieved by using two GPUs as this model is split into two pipelines. The use of ReLU function addresses the vanishing gradient issue, while leaving the output unbounded. To resolve the unbounded output issue AlexNet tries to normalize the output through a process called Local Response Normalization (LRN) as summarized in Table 3. The normalization operation is described as amplifying the

excited neurons while dampening the surrounding neurons in the local neighborhood. In AlexNet model, the overfitting problem is overcome using dropout regularization. Dropout regularization is a process where randomly selected neurons are dropped due to which the output from the corresponding nodes will not be passed to the subsequent nodes during forward pass. Similarly, the weights of the dropped neurons will not be updated during back pass. Due to this process the neural network learns multiple independent representations, thereby avoiding the overfitting problem (Wei, 2020).

One of the challenges with AlexNet model is that it contains more than 60 million parameters due to which the time to train might increase. Although the layers in AlexNet were able to better extract the features when compared to the CNN model that is trained from scratch, the layers were not deep enough when compared ResNet. This might result in the model facing challenges while learning features from certain type of image sets. There are filters of varying sizes that are employed by AlexNet in different convolution layers. Due to the use of filters with larger size, the number of parameters to be trained will increase and thereby increasing the training time (Wei, 2020).

ResNet-50 is another CNN model that is considered for this research project, since it has better accuracy and speed when compared to similar models. Due to the increase in number of layers in deep CNN, the model fails to learn new features at certain point which is called degradation problem. This problem is addressed with the use of residual networks in ResNet-50, where skip connections are introduced to skip the already learned features in the intermittent layers as summarized in Table 4. This approach will significantly improve the performance and reduce the training time. As ResNet-50 employs more layers with small kernel sizes, it increases the non-linearity of features which is an important aspect. Also, the use of residual blocks in

ResNet-50 results in limiting the validation losses due to which the problem of overfitting is avoided. ResNet-50 enables us to create very deep neural network model without the risk of degradation problem (Ayyar, 2020).

Table 2

Advantages and disadvantages of CNN model trained from scratch.

Model	Advantages	Disadvantages
CNN	Automatic detection of features. No need for manual intervention. Availability of large number of trainable parameters in various layers. Extraction of distinguishing features at different abstraction levels.	Requires immense volume of data for training the model. More computing power is needed. No definitive approach in determining layers. Not efficient when a smaller number of layers are involved.

Note. Advantages and disadvantages of CNN model trained from scratch.

Table 3

Advantages and disadvantages of AlexNet.

Model	Advantages	Disadvantages
AlexNet	Pre-trained model based on transfer learning. Use of two GPUs result in faster computation. Supports industry standard methods like ReLU, dropout, etc. Use of Local Response Normalization to resolve unbound output problem. Use of Dropout regularization to address overfitting problem	Limited depth when compared to other models. Increased training time due to high volume of parameters to train. Use of convolution filters that are larger in size. Inferior performance when compared to complex models like GoogleNet, ResNet etc.

Note. Advantages and disadvantages of AlexNet.

Table 4

Advantages and disadvantages of ResNet-50.

Model	Advantages	Disadvantages
ResNet-50	More number of layers results in effective understanding of features. Degradation problem is resolved with skip connection mechanism. Saves time and results in improved performance by skipping already learned features. Low validation loss due to use of residual networks.	Complex architecture. Batch normalization requires more computational resources. Without transfer learning this model could result in larger training times.

Note. Advantages and disadvantages of ResNet-50.

4.4 Model Evaluation Methods

It is crucial for any machine learning model to be trained to optimum level such that its performance is not just limited to the initial dataset that it was trained with but holds good for any new data under different scenarios. The stability of the model and its consistency in prediction with best accuracy across varied datasets are the factors that determine how well the model is built. To validate and evaluate the different models used in this project, the data from the transformation phase is categorized into training set and test set in 80% and 20% proportions. The models are trained, and the model predictions are tested using the train and test data that is afresh to the model respectively. The test data should be isolated until the model is trained and validated. Evaluation is done using the test data after the validation is completed. There are different techniques that are available to validate the stability of the model. Cross validation technique is used in this project to assess the ability of the model to work well with any new dataset in the future.

Cross validation techniques are classified as exhaustive and non-exhaustive. K-fold and stratified k-fold are the non-exhaustive cross validation techniques that are used in this project. One of the challenges with deep learning models is that they require huge data to train and validate. There is a chance that the model will be blindsided by some of the dominant patterns if enough data is not available to train the model during the training phase. The risk of reduction in accuracy due to error induced by bias, is a problem that is associated with providing the model with limited data during the training phase. Ample data is required to be provided to the model to avoid such drawbacks during the training phase. The data is divided into subsets of k in this validation technique. One out of k subsets is considered as a dataset to validate while the remaining subsets are used to train the model. The process is iterated for k iterations and error is captured for each iteration. The average error is calculated from the errors captured from all the iterations to ensure that each subset takes part in the validation set at least once. The errors induced by bias and variance were significantly reduced using this technique. Stratified K-Fold cross validation technique which is similar to K-Fold, is also employed in this project to prepare the dataset for validation. It differs from K-Fold technique in minor variation where instead of using random sampling, stratified sampling techniques are used. The mean responsive value is approximately equal in all the folds (Lakshana, 2021).

10-Fold cross validation technique is used for modified CNN model trained from scratch, 4-Fold cross validation technique is used for AlexNet and stratified 10-Fold cross validation technique is used for ResNet-50.

Dice similarity coefficient (DSC) is a metric that is employed to determine the U-Net model efficiency by finding out the similarity between U-Net generated samples and manually

annotated samples (Neha S, 2020). Mathematical representation of DSC is given in Equation 7 where G is truth data and S is Segmentation output.

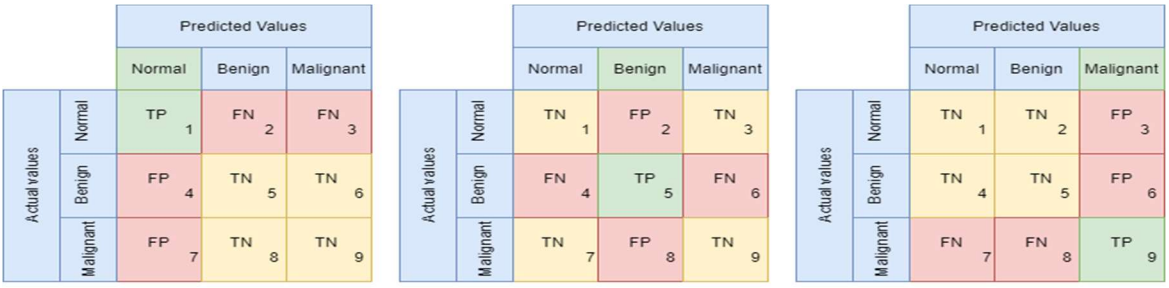
$$Dice\ Coefficient = 2(G \cap S) \tag{7}$$

Evaluation is the conclusive step to determine the performance of the model. Confusion Matrix, Validation Loss, Validation Accuracy, Generalization Gap and AUC-ROC curve are the different evaluation methods used across the three models in this project. Modified CNN model is evaluated using confusion matrix, validation loss, and validation accuracy. Confusion matrix is employed in evaluation of the modified CNN model to verify if different classes were predicted properly to their actual class.

Confusion matrix is an NxN matrix and since there are three target classes in this project, a 3x3 confusion matrix is used. The values predicted by the classifiers are represented in the columns and the actual values of the dataset are represented along the rows (Bharathi, 2021). Typically, the confusion matrix for a binary dataset consists of a simple 2x2 matrix with values as True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN) (Bharathi, 2021). Since the dataset used in this project is a multiclass, the corresponding matrix values have to be calculated.

Figure 41

Confusion Matrix for Normal, Benign and Malignant Classes



Note. Confusion Matrix for Normal, Benign and Malignant Classes.

From Figure 41, Confusion Matrix for three classes can be observed. In the Confusion Matrix for normal class, normal is considered as positive while benign and malignant are negative. Similarly for benign class, benign is considered as positive while normal and malignant being negative and for malignant class, malignant is considered as positive while normal and benign are negative.

The ratio of actual to predicted positive values is known as TP. Similarly, TN is the ratio of actual to the predicted negative values. Similarly, FP and FN values correspond to the rate at which actual negative values are positively predicted and the actual positive values are negatively predicted respectively. The four major metrics precision, recall, accuracy, and f- score from the confusion matrix are used in performance evaluation of the classification model. Accuracy is employed to identify the ratio of categorizing the values correctly into different classes, which is calculated as ratio of the sum of all true and total values as shown in Equation 8. Precision is utilized to determine the ability of the model in proper classification of positive values which is calculated as ratio of the number of true to predicted positive values as shown in Equation 9. Recall is used in computing the ability of the model to detect positive values which is calculated as the ratio of true to actual positives as shown in Equation 10. The numerical average between precision and recall is known as F1-Score, Mathematical representation of F1-Score is shown in Equation 11 (Bharathi, 2021).

$$Accuracy = \frac{TN + TP}{TN + FP + TP + FN} \tag{8}$$

$$Precision = \frac{TP}{TP + FP} \tag{9}$$

$$Recall = \frac{TP}{TP + FN} \tag{10}$$

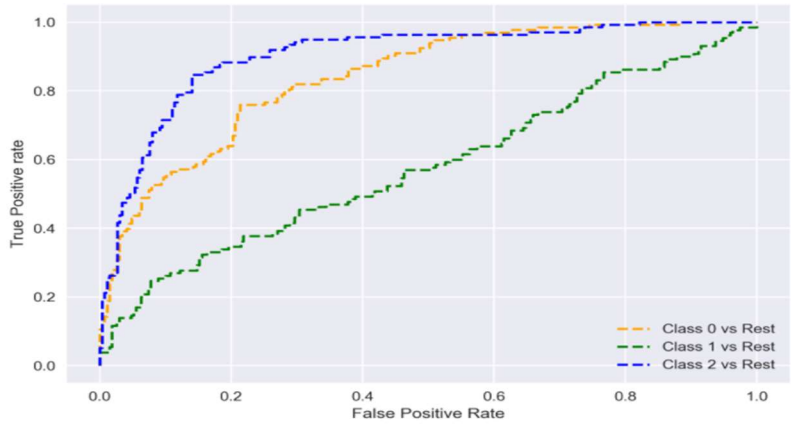
$$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (11)$$

After the dataset is segregated into training, test and validation sets during cross validation, the loss that is derived from the validation set is termed as validation loss. The model is evaluated to be efficient if the validation loss decreases with the increase in iterations of epochs of the model. If the validation loss is far greater than training loss, it results in an overfitting. The degree of overfitting depends on how far the validation loss differs from training loss. If the validation loss is less than training loss, then it results in underfitting and the degree of underfitting is based on how much less the validation loss is when compared to training loss.

The AlexNet model used in this project is evaluated from the metrics like precision, recall, accuracy, and F1-score which are computed by using Confusion Matrix. The AUC-ROC curve is also employed in this project to determine the efficiency of the model by its classification potential to distinguish between different images. Typically, AUC-ROC is used to measure binary classifications. Since this project has multi class classifications, one versus rest strategy is used to plot AUC-ROC curve for normal, benign, and malignant as shown in Figure 42. The AUC-ROC curve is evaluated depending on sensitivity and specificity of the image classes. Sensitivity is a measure to identify how many of the classes that are positive are exactly identified and specificity is a measure to identify the number of negative classes that were identified correctly (Bhandari, 2020). The AOC-ROC curve is a representation of plotting the sensitivity against one minus specificity at each possible cut-off. The model is expected to have better results if the ROC curve remains close to left most corner of the top quadrant (Bhandari, 2020).

Figure 42

Multiclass ROC Curve



Note. Multiclass ROC Curve. Adapted from "*AUC-ROC Curve in Machine Learning Clearly Explained*" by Bhandari, A. 2020, *Analytics Vidhya*, p.6 (https://www.analyticsvidhya.com/blog/2020/06/auc-roc-curve-machine-learning/#h2_8-AUC%20and%20ROC), Copyright 2020 by Analytics Vidhya.

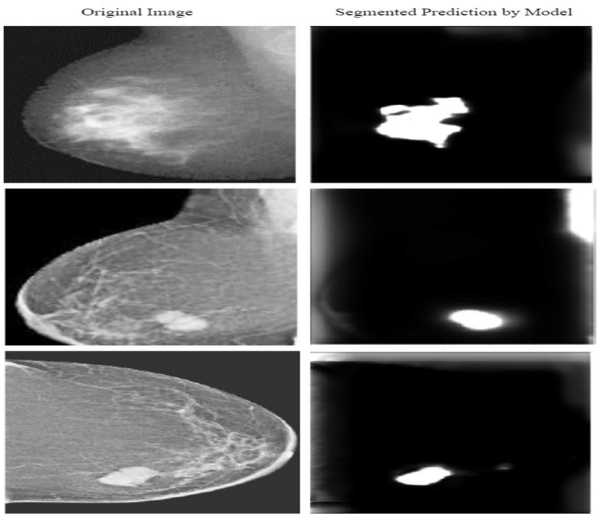
The ResNet-50 model used in this project is also evaluated similarly with the use of confusion matrix, validation loss, validation accuracy and AOC-ROC curve.

4.5 Model Validation and Evaluation

The different parameters used for implementing the U-Net model are 40 epochs with 300 steps per each epoch. Tesla V100 GPU is used for training the model which gave 0.96 dice coefficient (Neha S, 2020). The output of the U-Net model is the segmented map image as shown in Figure 43.

Figure 43

Original Mammogram Images and Segmented Output Images from U-Net



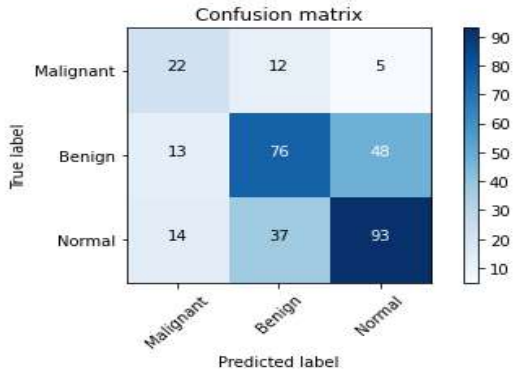
Note. Original mammogram images and segmented output images from U-Net Adapted from "Lesion Segmentation from Mammogram Images using a U-Net Deep Learning Network," by Neha S, 2020, *International Journal of Engineering Research & Technology (IJERT)*, *V9*(02), p. 4 (https://doi.org/10.17577/ijertv9is020213). Copyright 2020 by IJERT.

There are three different CNN models used in this project for early diagnosis of breast cancer. Once the models are built, they are trained using the train dataset. The Modified CNN model is trained for 100 epochs with initial learning rate 0.001, learning drop factor 0.3 and learning drop period 0.6. The model is then validated with 10-Fold cross validation technique to make sure that the model is well trained. The model prediction is then tested using test data and

evaluated from metrics like precision, accuracy, recall, F1-Score, validation loss and validation accuracy. The output from a similar project is used to compare the results of this CNN model.

Figure 44

Confusion Matrix for Modified CNN Trained from Scratch



Note. Confusion Matrix for Modified CNN Trained from Scratch. Adapted from "*Using Deep Learning for Mammography Classification | Analytics Vidhya*" by Poles, C. 2021, *Medium*, p.8 (https://medium.com/analytics-vidhya/using-convolutional-neural-networks-for-mammogram-classification-6e67ed4e0cad), Copyright 2021 by Medium.

From Figure 44, it can be observed that there are 93 normal, 76 benign and 22 malignant images which were correctly predicted. Similarly, the incorrectly predicted images are also high in number. Hence this model is estimated to have around 58% accuracy as shown in the Figure 45 below. So, this model cannot be ideally considered to detect the impacted images correctly. This model needs to be fine-tuned using hyperparameter tuning and increase the number epochs to improve the accuracy of prediction.

Figure 45

Classification Report for Modified CNN Trained from Scratch

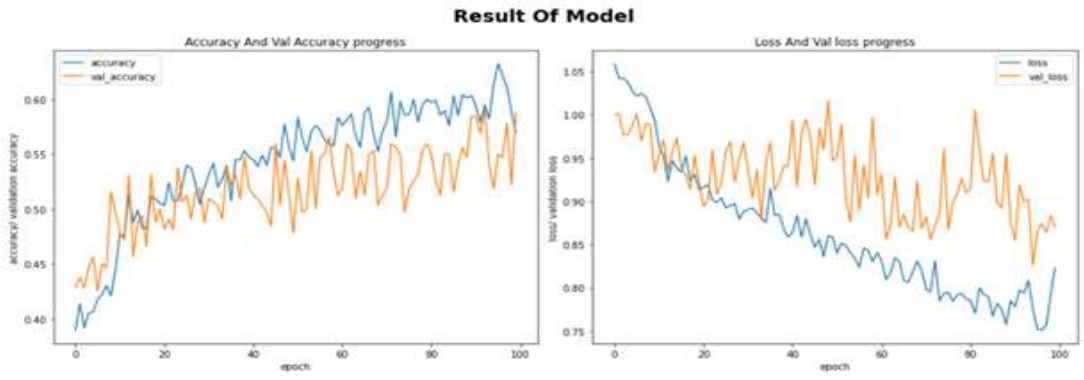
Classes: {'A': 0, 'B': 1, 'M': 2}

	precision	recall	f1-score	support
0	0.68	0.51	0.58	59
1	0.51	0.72	0.60	125
2	0.67	0.49	0.57	136
accuracy			0.58	320
macro avg	0.62	0.57	0.58	320
weighted avg	0.61	0.58	0.58	320

Note. Classification Report for Modified CNN Trained from Scratch. Adapted from "*Using Deep Learning for Mammography Classification | Analytics Vidhya*" by Poles, C. 2021, *Medium*, p.8. (https://medium.com/analytics-vidhya/using-convolutional-neural-networks-for-mammogram-classification-6e67ed4e0cad), Copyright 2021 by Medium.

Figure 46

Accuracy and Loss Plot for Training and Validation Data



Note. Accuracy and Loss Plot for Training and Validation Data. Adapted from "*Using Deep Learning for Mammography Classification | Analytics Vidhya*" by Poles, C. 2021, *Medium*, p.9. (https://medium.com/analytics-vidhya/using-convolutional-neural-networks-for-mammogram-classification-6e67ed4e0cad), Copyright 2021 by Medium.

From Figure 46, it can be observed that the graphs depict epochs versus training and validation accuracy and loss.

The AlexNet model is trained for 30 epochs with initial learning rates 0.0001 and 0.00001 (Hassan et al., 2020). The model is then validated using 4-Fold cross validation technique and the model prediction is then tested using test data and evaluated from different metrics like precision, accuracy, recall, F1-Score, validation loss and validation accuracy. The output from a similar project is used to compare the results of this AlexNet model.

The Adam optimizer is used to optimize the AlexNet model to increase the accuracy and minimize the loss with learning rate 0.00001, and eventually prevent it from the problem of overfitting. From Figure 47 it can be noticed that the gap between validation accuracy and training accuracy is very little, which proves that this model is not overfitted.

Figure 47

AlexNet Metrics for Learning rate 0.0001 and 0.00001

Model	Optimizer	LR	Training accuracy	Validation accuracy	Training loss	Validation loss	Generalization gap
(1) AlexNet	**Adam**	**1e-4**	96.65%	96.04%	0.0538	0.1785	0.1247
		1e-5	**97.26%**	**97.18%**	**0.0140**	**0.0662**	**0.0522**

Note. AlexNet Metrics for Learning rate 0.0001 and 0.00001. Adapted from "Breast Cancer Masses Classification Using Deep Convolutional Neural Networks and Transfer Learning" by Hassan et al., 2020, *Multimedia Tools and Applications*, *79*(41–42), 30735–30768, p.17. (https://doi.org/10.1007/s11042-020-09518-w), Copyright 2020 by Multimedia Tools and Applications.

Figure 48

AlexNet Validation Accuracy and Training Time for Each Fold

Fold Test	AlexNet	
	Validation accuracy	Training Time (30 epoch)
1st fold	**98.33%**	**8 min, 11 s**
2nd fold	**97.29%**	**8 min, 24 s**
3rd fold	**95.38%**	**8 min, 29 s**
4th fold	**97.71%**	**8 min, 16 s**
Average	**97.18%**	**8 min, 20 s**

Note. AlexNet Validation Accuracy and Training Time for Each Fold. Adapted from "Breast Cancer Masses Classification Using Deep Convolutional Neural Networks and Transfer Learning" by Hassan et al., 2020, *Multimedia Tools and Applications*, *79*(41–42), 30735–30768, p.23. (https://doi.org/10.1007/s11042-020-09518-w), Copyright 2020 by Multimedia Tools and Applications.

Figure 48 denotes validation accuracy for each fold and the corresponding time taken to validate the model. Validations from each fold are considered to calculate the average validation accuracy.

Figure 49

AlexNet Metrics Using Test Data for Each Fold

Fold Test	AlexNet				
	Sens.	Spec.	F1 - Score	MCC	ACC
1st fold	**95.63%**	**95.63%**	**95.63%**	**91.25%**	**95.63%**
2nd fold	**93.13%**	**99.38%**	**96.13%**	**92.69%**	**96.25%**
3rd fold	**96.88%**	**98.75%**	**97.79%**	**95.65%**	**97.82%**
4th fold	**86.13%**	**99.38%**	**92.31%**	**86.38%**	**92.82%**
Average	**90.47%**	**98.22%**	**95.47%**	**91.49%**	**95.63%**

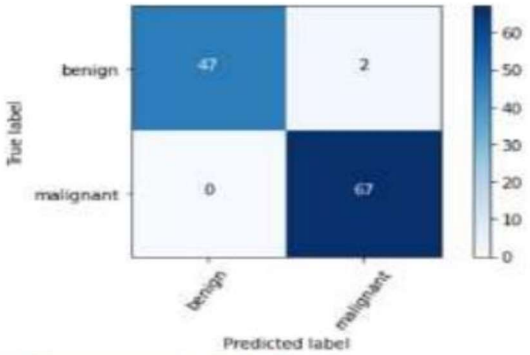
Note. AlexNet Metrics Using Test Data for Each Fold. Adapted from "Breast Cancer Masses Classification Using Deep Convolutional Neural Networks and Transfer Learning" by Hassan et al., 2020, *Multimedia Tools and Applications*, *79*(41–42), 30735–30768, p.23.(https://doi.org/10.1007/s11042-020-09518-w), Copyright 2020 by Multimedia Tools and Applications.

Figure 49 provides information about different metrics like sensitivity, specificity, accuracy etc. which denotes an overall accuracy of 95.63% that is considered as a good model for breast cancer detection. However, the accuracy can be still improved by using advanced CNN models with more layers which can better extract features.

The ResNet-50 model is trained for a maximum of 20 epochs. The model is then validated using stratified 10-Fold cross validation technique. The model is then tested for prediction using test data and evaluated from different metrics like precision, accuracy, recall, Area under the ROC curve, and F-Score. The output from a similar project is used to compare the result of this model.

Figure 50

Confusion Matrix for ResNet-50



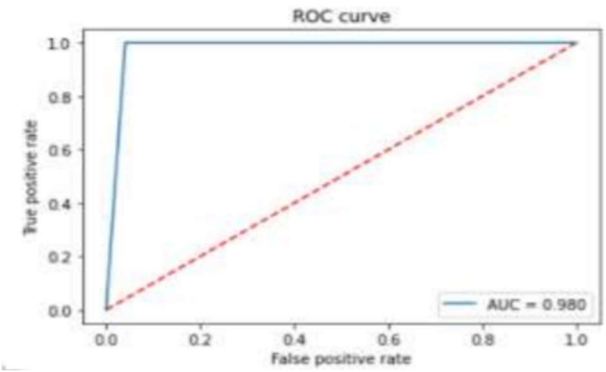
Note. Confusion Matrix for ResNet-50. Adapted from "Breast Cancer Prediction using ResNet-50" by Prasanna, M., Thamarai, M., & Malarvizhi, S. P. 2020, *International Journal of Advanced Research Trends in Engineering and Technology (IJARTET), 7*(9), p.6. (https://doi.org/10.20247/IJARTET.2020.0709004), Copyright 2020 by IJARTET.

From the Figure 50, it can be observed that 67 malignant and 47 benign cases were correctly predicted and only two cases of benign type were incorrectly predicted as malignant

and no cases of malignant were wrongly predicted. The precision was estimated at 97%, recall as 100%, F-Score as 99% and accuracy as 98.6% for the ResNet-50 model (Prasanna et al., 2020).

Figure 51

Area Under ROC Curve for ResNet-50



Note. Area Under ROC Curve for ResNet50. Adapted from "Breast Cancer Prediction using ResNet-50" by Prasanna, M., Thamarai, M., & Malarvizhi, S. P. 2020, *International Journal of Advanced Research Trends in Engineering and Technology (IJARTET), 7*(9), p.7. (https://doi.org/10.20247/IJARTET.2020.0709004), Copyright 2020 by IJARTET.

This model is tuned for 20 epochs. Ideally if the AUC value turns out to be 1.0, the model is considered to predict all the classifications correctly. This AUC value for this model calculated from the curve is 0.98 as shown in the Figure 51 (Prasanna et al., 2020). The greater the AUC value, the better is the model and since this model has an AUC 0.98 value, it evaluates that this model is successful in classifying the majority of mammographic images from the dataset. After comparing the performance metrics of the three proposed models, it is evident that ResNet-50 turns out to be the best performing model.

References

Acr, R. A. (2021, February 8). *Mammography*. Radiologyinfo.Org. https://www.radiologyinfo.org/en/info/mammo

Alanazi, S. A., Kamruzzaman, M. M., Islam Sarker, M. N., Alruwaili, M., Alhwaiti, Y., Alshammari, N., & Siddiqi, M. H. (2021). Boosting Breast Cancer Detection Using Convolutional Neural Network. *Journal of Healthcare Engineering*, *2021*, 1–11. https://doi.org/10.1155/2021/5528622

Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M., & Farhan, L. (2021). Review of Deep Learning: Concepts, CNN Architectures, Challenges, Applications, Future Directions. *Journal of Big Data*, *8*(1). https://doi.org/10.1186/s40537-021-00444-8

Amazon S3 Replication. (n.d.). Amazon Web Services, Inc. Retrieved October 9, 2021, from https://aws.amazon.com/s3/features/replication/

Arc. (2018, December 26). *Convolutional Neural Network - Towards Data Science*. Medium. https://towardsdatascience.com/convolutional-neural-network-17fb77e76c05

Aslan, M. F., Celik, Y., Sabanci, K., & Durdu, A. (2018). Breast Cancer Diagnosis by Different Machine Learning Methods using Blood Analysis data. *International Journal of Intelligent Systems and Applications in Engineering*, *6*(4), 289–293. https://doi.org/10.18201/ijisae.2018648455

AWS Command Line Interface. (n.d.). Amazon Web Services, Inc. Retrieved October 9, 2021, from https://aws.amazon.com/cli/

AWS Pricing Calculator. (n.d.). Retrieved October 8, 2021, from https://calculator.aws/#/

Ayyar, T. M. (2020, November 6). *A Practical Experiment for Comparing LeNet, AlexNet, VGG and ResNet Models with Their Advantages and Disadvantages.* Medium. https://tejasmohanayyar.medium.com/a-practical-experiment-for-comparing-lenet-alexnet-vgg-and-resnet-models-with-their-advantages-d932fb7c7d17

Bhandari, A. (2020, July 20). *AUC-ROC Curve in Machine Learning Clearly Explained*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2020/06/auc-roc-curve-machine-learning/#h2_8-AUC%20and%20ROC

Bharathi (2021, July 16). *Confusion Matrix for Multi-Class Classification*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/06/confusion-matrix-for-multi-class-classification/

Charan, S., Khan, M. J., & Khurshid, K. (2018). Breast Cancer Detection in Mammograms Using Convolutional Neural Network. *International Conference on Computing, Mathematics and Engineering Technologies* (ICOMET),17748987, https://doi.org/10.1109/ICOMET.2018.8346384

Devakumari, D., & Punithavathi, V. (2018). Comparison of Noise Removal Filters for Breast Cancer Detection in Mammogram Images. *International Journal of Pure and Applied Mathematics*, 18, 3863-3874. https://acadpubl.eu/hub/2018-119-18/3/312.pdf

Elter, M., & Horsch, A. (2009). CADx of Mammographic Masses and Clustered Microcalcifications: A Review. *Medical Physics*, *36*(6Part1), 2052–2068. https://doi.org/10.1118/1.3121511

Gupta, D. (2020, July 19). *Activation Functions | Fundamentals of Deep Learning*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2020/01/fundamentals-deep-learning-activation-functions-when-to-use-them/

Hamed, G., Marey, M., Amin, S. E. S., & Tolba, M. F. (2018). A Proposed Model for Denoising Breast Mammogram Images. *2018 13th International Conference on Computer Engineering and Systems (ICCES)*. Published. https://doi.org/10.1109/icces.2018.8639307

Hassan, S. A., Sayed, M. S., Abdalla, M. I., & Rashwan, M. A. (2020). Breast Cancer Masses Classification Using Deep Convolutional Neural Networks and Transfer Learning. *Multimedia Tools and Applications*, *79*(41–42), 30735–30768. https://doi.org/10.1007/s11042-020-09518-w

Hien, D. H. T. (2018, June 20). *A Guide to Receptive Field Arithmetic for Convolutional Neural Networks*. Medium. https://blog.mlreview.com/a-guide-to-receptive-field-arithmetic-for-convolutional-neural-networks-e0f514068807

Introduction to Amazon S3. (n.d.). Amazon Web Services, Inc. Retrieved October 9, 2021, from https://aws.amazon.com/s3/

Introduction to Amazon S3 Glacier. (n.d.). Amazon Web Services, Inc. Retrieved October 9, 2021, from https://aws.amazon.com/s3/glacier/

Jaamour, A., Patel, A., & Jen Chen, S. (2020, August). *Breast Cancer Detection in Mammograms using Deep Learning Techniques*. University of St Andrews- School of Computer Science. https://doi.org/10.5281/zenodo.3985051

Jimenez-Gaona, Y., Rodriguez-Alvarez, M. J., & Lakshminarayanan, V. (2020). Deep-Learning-Based Computer-Aided Systems for Breast Cancer Imaging: A Critical Review. *Applied Sciences*, *10*(22), 8298. https://doi.org/10.3390/app10228298

Kaushik, A. (2020, July 21). *Understanding ResNet50 Architecture*. OpenGenus IQ: Computing Expertise & Legacy. https://iq.opengenus.org/resnet50-architecture/

Koech, K. E. (2021, October 5). *Cross-Entropy Loss Function - Towards Data Science*. Medium. https://towardsdatascience.com/cross-entropy-loss-function-f38c4ec8643e

Lakshana, V. (2021, May 27). *Cross-Validation Techniques in Machine Learning for Better Model*. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/05/4-ways-to-evaluate-your-machine-learning-model-cross-validation-techniques-with-python-code/

Li, H., Zhuang, S., Li, D.-ao, Zhao, J., & Ma, Y. (2019). Benign and Malignant Classification of Mammogram Images Based on Deep Learning. *Biomedical Signal Processing and Control*, *51*, 347–354. https://doi.org/10.1016/j.bspc.2019.02.017

Liu, Z., Dou, Y., Jiang, J., & Xu, J. (2016) Automatic Code Generation of Convolutional Neural Networks in FPGA Implementation. *International Conference on Field-Programmable Technology (FPT)*, 2016, https://doi.org/0.1109/FPT.2016.7929190

Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., van der Laak, J. A., van Ginneken, B., & Sánchez, C. I. (2017). A Survey on Deep Learning in Medical Image Analysis. *Medical Image Analysis*, *42*, 60–88. https://doi.org/10.1016/j.media.2017.07.005.

Manna, S. (2020, June 26). *K-Fold Cross Validation for Deep Learning Models using Keras*. Medium. https://medium.com/the-owl/k-fold-cross-validation-in-keras-3ec4a3a00538

Michael, E., Ma, H., Li, H., Kulwa, F., & Li, J. (2021). Breast Cancer Segmentation Methods: Current Status and Future Potentials. *BioMed Research International*, *2021*, 1–29. https://doi.org/10.1155/2021/9962109

MurtiRawat, R., Panchal, S., Singh, V. K., & Panchal, Y. (2020). Breast Cancer Detection Using K-Nearest Neighbors, Logistic Regression, and Ensemble Learning. *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*. https://doi.org/10.1109/icesc48915.2020.9155783

Neha S, T. (2020). Lesion Segmentation from Mammogram Images using a U-Net Deep Learning Network. *International Journal of Engineering Research & Technology (IJERT)*, *V9*(02). https://doi.org/10.17577/ijertv9is020213

Omondiagbe, D. A., Veeramani, S., & Sidhu, A. S. (2019). Machine learning classification techniques for breast cancer diagnosis. *IOP Conference Series: Materials Science and Engineering*, *495*, 012033. https://doi.org/10.1088/1757-899x/495/1/012033

Overview of Amazon EC2 P3 Instances. (n.d.). Amazon Web Services, Inc. Retrieved October 9, 2021, from https://aws.amazon.com/ec2/instance-types/p3/

Ponraj, D., Jenifer, M., Poongodi, P., Manoharan, Samuel. (2011). A Survey on the Preprocessing Techniques of Mammogram for the Detection of Breast Cancer. *Journal of Emerging Trends in Computing and Information Sciences*, 12, 2079-8407. https://www.researchgate.net/publication/266489422_A_Survey_on_the_Preprocessing_Techniques_of_Mammogram_for_the_Detection_of_Breast_Cancer

Poles, C. (2021, July 15). *Using Deep Learning for Mammography Classification | Analytics Vidhya*. Medium. https://medium.com/analytics-vidhya/using-convolutional-neural-networks-for-mammogram-classification-6e67ed4e0cad

Pricing. (2021, June 4). Draw.Io. https://drawio-app.com/pricing/

Prasanna, M., Thamarai, M., & Malarvizhi, S. P. (2020). Breast Cancer Prediction using ResNet50. *International Journal of Advanced Research Trends in Engineering and Technology (IJARTET*, *7*(9). https://doi.org/10.20247/IJARTET.2020.0709004

Rawat, A. S. (n.d.). *Importance of Statistics for Data Science | Analytics Steps*. https://www.analyticssteps.com/blogs/importance-statistics-data-science

Salama, W. M., & Aly, M. H. (2021). Deep Learning in Mammography Images Segmentation and Classification Automated CNN Approach. *Alexandria Engineering Journal*, *60*(5), 4701–4709. https://doi.org/10.1016/j.aej.2021.03.048

Sarkar, D. (2018, November 17). *A Comprehensive Hands-on Guide to Transfer Learning with Real-World Applications in Deep Learning*. Medium. https://towardsdatascience.com/a-comprehensive-hands-on-guide-to-transfer-learning-with-real-world-applications-in-deep-learning-212bf3b2f27a

Shen, L., Margolies, L. R., Rothstein, J. H., Fluder, E., McBride, R., & Sieh, W. (2019). Deep Learning to Improve Breast Cancer Detection on Screening Mammography. *Scientific Reports*, *9*(1). https://doi.org/10.1038/s41598-019-48995-4

Solai, P. (2020, June 22). *How does Backpropagation work in a CNN?* Medium. https://pavisj.medium.com/convolutions-and-backpropagations-46026a8f5d2c

Suckling, J., Parker, J., Dance, D., Astley, S., Hutt, I., Boggis, C., Ricketts, I., Stamatakis, E., Cerneaz, N., Kok, S., Taylor, P., Betal, D., Savage, J. (2015). *Mammographic Image Analysis Society (MIAS) database v1.21* [Dataset]. https://www.repository.cam.ac.uk/handle/1810/250394

Talasila, A. (2021, January 21). *Generating Image Segmentation Masks — The Easy Way - Towards Data Science*. Medium. https://towardsdatascience.com/generating-image-segmentation-masks-the-easy-way-dd4d3656dbd1

Tripathy, S., & Swarnkar, T. (2020). Unified Preprocessing and Enhancement Technique for Mammogram Images. *Procedia Computer Science*, *167*, 285–292. https://doi.org/10.1016/j.procs.2020.03.223

U.S. Breast Cancer Statistics. (2021, February 4). Breastcancer.Org. https://www.breastcancer.org/symptoms/understand_bc/statistics

Wei, J. (2020, September 25). *AlexNet: The Architecture that Challenged CNNs - Towards Data Science*. Medium. https://towardsdatascience.com/alexnet-the-architecture-that-challenged-cnns-e406d5297951

Zhang, L., Gao, H. J., Zhang, J., & Badami, B. (2020). Optimization of the Convolutional Neural Networks for Automatic Detection of Skin Cancer. *Open Medicine*, *15*(1), 27–37. https://doi.org/10.1515/med-2020-00