Skin-Cancer Classification Using Deep Learning with DenseNet and VGG with Streamlit-Framework Implementation

MSc Research Project
Data Analytics

Raj Shrikant Sonawane
Student ID: x211555054

School of Computing
National College of Ireland

Supervisor: Prof. Noel Cosgrave

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Raj Shrikant Sonawane
Student ID:	x21155054
Programme:	MSc Data Analytics
Year:	2022-2023
Module:	MSc Research Project
Supervisor:	Prof. Noel Cosgrave
Submission Due Date:	15/12/2022
Project Title:	Skin-Cancer Classification Using Deep Learning with DenseNet and VGG with Streamlit-Framework Implementation
Word Count:	6317
Page Count:	21

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	
Date:	30th January 2023

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Skin-Cancer Classification Using Deep Learning with DenseNet and VGG with Streamlit-Framework Implementation

Raj Shrikant Sonawane
x21155054

Abstract

Skin cancer is consistently ranked among the top three most lethal tumors that are caused by DNA damage and can lead to death on a worldwide scale. The growth of the epidermal layer of the skin is abnormal or rapid, which encourages the formation of tumors in the body. After being identified, this has had an effect on people of all ages and has a reputation for being an expensive ailment to treat in the past. An earlier study that looked at the various types of skin cancer found that earlier detection might result in quicker treatment and lower overall costs. This research project aims to build Deep Learning models that will assist in classifying skin lesions as either benign or malignant. The models will be used in the context of a skin cancer diagnosis. The application of deep learning will help cut both the amount of time and money spent on therapy. Using the Streamlit application development framework, a web-based application has been developed with the intention of bridging the gap between Deep Learning and medical expertise. The dataset compiled by the International Skin Image Collaboration and hosted on Kaggle has been utilized for the purpose of training and testing algorithmic models. There was no evidence of class imbalance in the data; however, space limitations necessitated the application of a variety of Data Augmentation strategies. For categorization, deep learning models including DenseNet-121, DenseNet-169, DenseNet-201, VGG-16, and VGG-19 have been created. By considering and striking a balance between the different metrics that are necessary in the field of medicine, DenseNet-121 with three layers was able to achieve scores of 98.24 and 70.55 for its sensitivity and specificity, respectively, and 84.64 for its accuracy. The findings demonstrated that models' performance can be improved by adjusting their hyper-parameters.

1 Introduction

Skin cancer is one of the most dangerous types of cancer. Skin cancer is caused by DNA mistakes in skin cells and can also be caused by exposure to sunshine. The need for early detection of skin cancer symptoms is necessitated by the increasing incidence of the disease, the high mortality rate, and the high cost of treatment. In light of the severity of these issues, researchers have developed a variety of early detection techniques for skin cancer that will facilitate treatment and save costs. This research can identify skin cancer and distinguish benign skin cancer from melanoma based on lesion characteristics such as symmetry, color, size, shape, etc. Produced by detection and classification technologies Techniques and methods for data analysis that may be beneficial to Medical Stakeholders such as Doctors, Patients, and Insurance Company policymakers. Through a comprehensive literature analysis on the diagnosis of skin cancer through image classification algorithms, this concept will be implemented as a thesis project.

1.1 Research Question

How successfully and accurately can Deep Learning models be developed using parameter tuning techniques to classify if a person has skin cancer deliver the classification model by developing a web-based application?

1.2 Motivation and Objectives

Healthcare systems now struggle with inefficiency or a lack of understanding of how to use predictive models effectively. The proposed thesis study would develop models using performance-tuning techniques and establish a web-based application for the model in order to understand the workings of the Deep Learning model. The development or rebuilding of previous models, as well as the use of Hyper-Parameter tuning, Transfer Learning Techniques, and the creation of a Web application utilizing Streamlit, are all included in the work for this thesis. With the outcomes of this thesis project, it will be simpler to comprehend the stakeholders and ensure the appropriate use of data analysis methodologies. In order to support the theories and draw inspiration from various models developed by other studies in the medical science sector dealing with image processing, this paper provides a thorough systematic evaluation of deep learning methods for the early diagnosis of skin cancer, such as DenseNet and VGG. The method utilized to address the query of why such approaches are chosen is described in the methodology section. The use of the web application interface known as the Streamlit framework is also suggested as an experimental choice for classification models. Evaluation standards that are generally accepted in the field of medical science will be used to determine how accurately the categorization model is built.

2 Related Work

To categorize whether skin cancer is malignant or not, many machine learning, deep learning, and other prediction algorithms must be utilized. The best performing techniques could be selected depending on the pertinent assessment criteria. Diverse image processing approaches have been used in numerous research studies to classify skin lesions using image datasets from various sources. This section will highlight the appropriate methods used by prior studies that can serve as an example or support for the work to be done in this project.

The literature review will be broken up into sections such as Why Is Prediction Necessary in Medical Science? Skin cancer prediction utilizing neural networks and deep learning, Exploratory data analysis in image processing and techniques for enhancing the performance of prediction models.

2.1 Why is prediction necessary in Medical Science?

The healthcare industry made significant strides as a result of new computer breakthroughs. This industry was pressured to offer knowledge that was more and more generative, which led to the emergence of numerous academic specialties. There are numerous efforts undertaken to both adapt to the deluge of medical information and learn anything meaningful from it Kaur et al. (2021). This inspired specialists to apply all the specialized technologies, including machine learning, deep learning, artificial intelligence (AI), learning algorithms, and predictive analytics, to assist in judgment and extract valuable knowledge. Prediction and classification models are used in medical research to evaluate the possibility that a disease will manifest so that it could be treated or avoided before it happens.

According to study on cancer's effects, tumor cancer cells may spread. Early diagnosis boosts survival odds for any illness, including cancer. It reduces the likelihood that benign tumors would require expensive treatment, allowing doctors to provide the best care and many researchers used Machine Learning to make a point. The research was limited to text-based data, and X-ray images would have improved it. Unsupervised learning has enhanced image processing prediction results. Medical imaging work allows radiologists to predict using visual patterns. X-rays, ECGs, MRI scans, etc. Unsupervised learning methods show potential for biological picture interpretation. Data Analytical applications include fMRI clustering analysis. Shehab et al. (2022) collected a large amount of data and used clustering to isolate brain health disorders using MRI for medical imaging. This work may aid in early diagnosis. According to a prediction and classification study, machine learning and deep learning offer cost-saving benefits in the medical business, including early detection and treatment. The Data Analytical sector of medicine is still expanding and faces many hurdles, but continued research will help overcome them by using the right data and methodologies.

2.2 Skin cancer prediction utilizing neural networks and deep learning

To ascertain whether skin cancer is malignant or not, this thesis work will be conducted among the various medical science fields. It is necessary to use a range of machine learning, deep learning, and other prediction algorithms, and the best ones can be chosen based on the relevant assessment criteria. In this thesis, image classification algorithms will be used to classify skin cancers using the necessary image datasets. Certain study in this area has produced some desired outcomes that can be used as learning materials for this thesis project.

Using similar or the same dataset for research may make it easier to make notes on different classification strategies. The HAM10000 Skin Cancer dataset from Kaggle, which had more than 10,000 photos of different types of skin cancer, was used in some research articles. Each paper employed a different categorization technique.

Researchers categorized skin cancer into 7 phases Labde and Vanjari (2022) using a variety of characteristics, including asymmetrical shape, diameters, border irregularities, colors, and evolution. Initially, it was suggested to employ MobileNet and Transfer Learning to achieve high accuracy. Since there were so many photos collected, the authors used a graphics processing unit to train and run their model. The 23 layers in their model were employed, and the optimizer was set to "Adam" with 30 epochs. They got about

85% accuracy out of their model. The creation of an application that allows users to upload a single image to categorize the type of skin cancer is an intriguing development that might serve as inspiration. However, since obtaining the dataset is easier and GPU compute capacity is almost comparable, it would have been preferable to run the model training procedure on Kaggle Notebook instead of Google Colab.

Some researchers deployed four distinct deep learning techniques using the same dataset from Kaggle repositories: recurrent neural network, convolutional neural network, Xception, and resnet50 Singh et al. (2022). By drawing a distribution plot, the data imbalance was found in a straightforward and practical manner. The author used ImageDataGenerator to implement data augmentation, and different Keras Library functions were employed when developing the model, which served as motivation for the current thesis work. Since the hyper-parameters were based on several trials and errors, Xception beat the other deep learning models listed. A genius strategy to increase the model's performance was to introduce the Xgboost algorithm.

For educational purposes, a dataset with 5 kinds of skin was by merging data photos from multiple sources that were freely available, and images were converted to 227*227 pixels and then deployed to a pre-trained model using the AlexNet architecture model in order to extract the necessary features Hameed et al. (2018). The model was used to create the ECOC linear SVM to categorize different skin types. The achieved accuracy was almost 86 percent. However, performance may have been improved by using various parameters.

Another method to develop a good model is to select the characteristics that are most informative. This was done by using the right visualization techniques to discover the aspects that are most informative. Such a method has been used to classify skin cancer using models including CNN, Artificial Neural Networks, and Transfer Learning, with ANN and Transfer Learning performing best [Kumar et al. (2022). Age, gender, cell type, and habitat were also taken into account. However, if the right epochs had been chosen, this study project could have been more productive.

The same feature selection technique was used to extract variables including intensity, texture, and color from a dataset obtained from a private medical institute Sharma and Chadha (2022). The algorithms employed were RNN, SVM, and Decision Tree. The model was flawed because it lacked suitable parameters and cross-validation. Anisotropic diffusion and Image Sharpening Using Masking Technique, however, stood out as the most prominent pre-processing approaches.

Multiple skin illnesses were included in a different private and genuine picture collection that the International Skin Imaging Collaboration acquired Ramachandro et al. (2021). The researchers utilized under-sampling and over-sampling techniques to address the severely unbalanced data. Following transformation, DenseNet was put into use, together with the necessary DenseBlock and Dense Layers. CNN was the next model to be constructed, and it had layers with maximum pooling configurations. While Deep Learning Models produced the greatest results, SVM and Random Forest machine learning models were also used.

2.3 Exploratory Data Analysis and Data Augmentation in Image Classification:

Data pre-processing or transformation techniques are necessary for every machine learning, deep learning, or neural network methodology in order to provide the model with relevant datasets. The methods of data analysis and data pre-processing used by earlier writers in a related subject will be highlighted in this section.

Large image datasets contain raw data that needs to be converted. Data transformation, dimensionality reduction, and feature selection were applied to the dataset used in [19]. Noises and a few undesirable artifacts were eliminated to get a good classification rate. Gaussian and median blurring were used to reduce noise. Data was reduced using dimensionality reduction techniques because the data had a huge volume. By using data uprightness and managing some anomalies, data normalization was accomplished.

A CNN model was developed by several researchers and professional hospital professionals to classify chest X-rays Jahan et al. (2021). Since the authors' data was so severely unbalanced, it had to be corrected before a model could be developed. To expand the size of the images for this purpose, data augmentation techniques were applied to the dataset. They were able to over-fit and resize the photos to 128*128 using this approach. They have recalculated the 'class weights' parameter to address the class imbalance. Although the model produced positive results, the researchers failed to provide the proper justification for data augmentation prior to handling class imbalance.

A recent study shows that deep learning-based image categorization algorithms can enhance accuracy with high-quality datasets. Current techniques solely consider classification accuracy, not dataset quality. ImageDC is a deep neural network-based image data cleaning framework Zhang et al. (2020). ImageDC was used to remove rare class images and noisy data to increase the datasets' recognition rate. Cleaning Evaluation, Cleaning with a Low Recognition Rate, and Cleaning with Minority Rate were the three factors used by these authors.

To test the viability of segmented liquor picking using hops pictures, an intelligent liquor picking system was created Li et al. (2022). The camera's frame rate is set to 25 and photos are transformed to sequence images by frame. To create a high-quality HD image, an image must first go through the image preprocessing steps of ROI region selection, Gauss denoising, image sharpening (USM Sharpening), and finally image brightness adjustment.

2.4 Image Classification using Deep Learning Models

The use of picture identification technology based on convolutional neural networks in species used in routine agricultural production has substantially helped the categorization and identification of agricultural items Feng et al. (2022). A convolution neural network was utilized to develop a model for quickly classifying crop ailments for the study project. The data sets of obtained crop disease images are trained using the DenseNet-201 network structure. The neurons in the top layers of the brain may have been tuned to increase accuracy, according to this research.

The authors' model, which concentrated on ResNet variants Sarwinda et al. (2021), SGD hyper-parameters, optimum function, and learning rate, reached a 96% sensitivity rate for the detection of colorectal cancer. By adjusting convolution layers with numerous trails, ResNet-18 and ResNet-50 were both tuned.

Another Deep Learning technique used in the field of medicine Majib et al. (2021) demonstrated the use of VGG16, which is based on the weights of an ImageNet dataset with 14 million images. This strategy may be helpful for this research because the pre-trained model has been trained on a large number of photos. The VGG-16 and VGG-19 models have excelled among all those created.

In keeping with medical research, researchers used early stopping and dropout layers to avoid over-generalization and used the sigmoid function as the class was dichotomous in order to detect coronavirus in lungs using VGG and DenseNet Hamwi and Almustafa (2022). As the classes are split into benign and malignant, this can serve as inspiration for current research work. In a study similar to this one, X-ray images of healthy and convalescent lungs were identified using ResNet18, VGG-16, and DenseNet-121, with DenseNet-121 achieving the highest accuracy scores Wang et al. (2021). The study used the Adam Optimizer and ReLu function but did not explain why it was used. All deep learning models were treated in the same way. The number of epochs utilized affected each model's accuracy.

2.5 Implementation of Streamlit Framework

Only someone with expertise in that field can interpret any machine learning or deep learning model's mechanism. A User Interface of a Web-based application may be useful to enable the outside world to utilize the models built.

Using machine learning algorithms to recognize faces behind masks was one of the essential use cases that needed to be addressed on a worldwide scale. When developing a Web-based application, the author made use of several different packages, including Stream light, OpenCV, TensorFlow, NumPy, and Keras Yalamarthi et al. (2022). The series of explanations that were given, such as loading photos to 224 by 224 using functions provided by Keras rather than using Numpy to convert the images into an array, were really helpful. This method will serve admirably as a source of motivation for the work that will be done on the research.

Another use of the Streamlit framework has recently been put into use for the detection of fraudulent credit card activity Jain et al. (2022). Only the weights of those models have been employed by the researchers, and all of those models had to reach an accuracy of 95% or higher. The build of the app had three different machine learning algorithms that users may choose from as an available option to select.

2.6 Conclusion

The investigation carried out on the aforementioned research work has shed light on a number of helpful methodologies, such as the application of DenseNet, VGG, ResNet, SVM models etc. The author has been motivated to conduct additional research by the ongoing works of their investigation. Both the parameters that were utilized throughout the construction of the model and the reasoning that was presented have contributed to how the architecture of the model might be improved. However, some of the study efforts may have failed to meet expectations in terms of model construction, but the researchers' data augmentation and exploratory data analysis were spots on. This research will implement an execution environment, data handling procedure, and developing a deep learning model while taking into consideration the parameters and configurations that will be used. This research will combine all of the previous learnings. The aforementioned study will serve as a significant source of motivation for the creation of the Steamlit Framework.

3 Methodology

The results of this research will assist medical professionals in the categorization of skin lesion kinds and the enhancement of treatment success rates. Medical Stakeholders like Doctors, Patients, and Insurance Company Policymakers may benefit from Classification Technologies that are Produced by Deep Learning and Transfer Learning techniques and tools along with appropriate User Interference. The technique known as CRISP-DM, which stands for "Cross-Industry Standard Process for Data Mining," will therefore be applied to this research endeavor.

CRISP-DM comprising of six phases will be described throughout this part which is implemented for this study endeavor.

3.1 Data Understanding

The dataset used in this study is called 'Skin Cancer: Malignant vs. Benign.' It was acquired from public repositories on Kaggle [1] and given that name. When conducting research in the field of medical science, one of the most important and difficult tasks is gathering photographs from reliable and official sources that can be used for investigational reasons. The International Skin Imaging Collaboration is the original source of the data that was retrieved from Kaggle. The data is organized into two separate folders labeled "Test" and "Train," and both folders include 1800 images. Each folder contains an additional two folders, which are divided into Benign and Malignant categories respectively. The figure that follows 1 provides an example of both benign and malignant images derived from the train and test data.

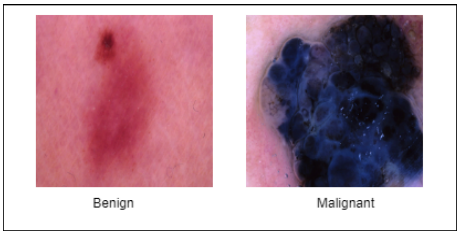

Figure 1: Sample Images

3.2 Data Preparation

The subsequent sequence of procedures was carried out in order to prepare the data.

3.2.1 Data Cleaning

The image dataset was downloaded from Kaggle repositories. The metadata associated with the datasets which is provided by the ISIC, is examined to determine whether or not it is missing any values and whether or not the directory contains any files. The obtained dataset does not contain any missing values or values that are null, and this is consistent for both the train and test folders.

[1] https://www.kaggle.com/datasets/fanconic/skin-cancer-malignant-vs-benign

3.2.2 Exploratory Data Analysis

In this part of the article, the author has carried out exploratory data analysis in order to determine how the classification is broken down. The performance of the classification model would be negatively impacted by the presence of imbalanced data, which could lead to biased output or low accuracy scores. The ratios in the testing data are the same as those in the training dataset. Visually representing the facts was the most efficient technique to determine the balance between them. In order to accomplish this, the author created a bar graph to display the distribution of the classifications. As can be seen in the figure 2 and 3, benign skin cancers make up a total of 55% of the data for both the train and test sets, whilst malignant skin cancer images make up the remaining 45%.

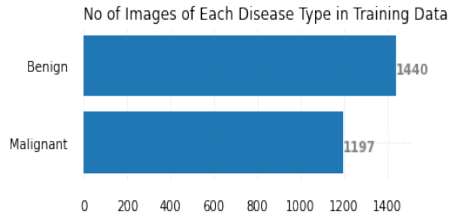

Figure 2: Class Balance for Train data

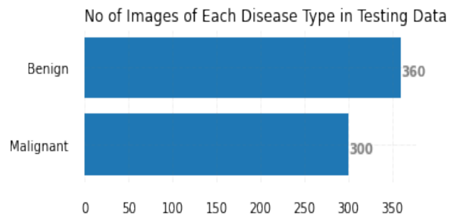

Figure 3: Class Balance for Test data

3.2.3 Data Augmentation

In this section, author has justified the implementation and necessity of Data Augmentation. With the given image dataset, there was an instance of overfitting. Data Augmentation is hence one of the strategies that may be utilized in order to enhance the amount of the dataset. The dataset is given more variation as a result of using this strategy and can create additional data from a single image data. The primary goal of boosting the size of the dataset is to reduce the likelihood of engaging in overfitting. The ImageDataGenerator module was utilized so that data augmentation could be carried out. Python's ImageDataGenerator is a notion for passing input to a model, as well as describing data augmentation strategies that can be used to enhance the size of the training data set. The rotation range was set to 10 and rescale values were kept 1./255. Shear and Zoom ranges were set to 0.2 and Horizontal Flip was enabled. Width Shift range and Height Shift Range were also kept to 0.2. The values were based on data augmentation techniques followed on Ali et al. (2021) where in similar problem was solved. This enabled us to overcome the overfitting issue as more images were generated.

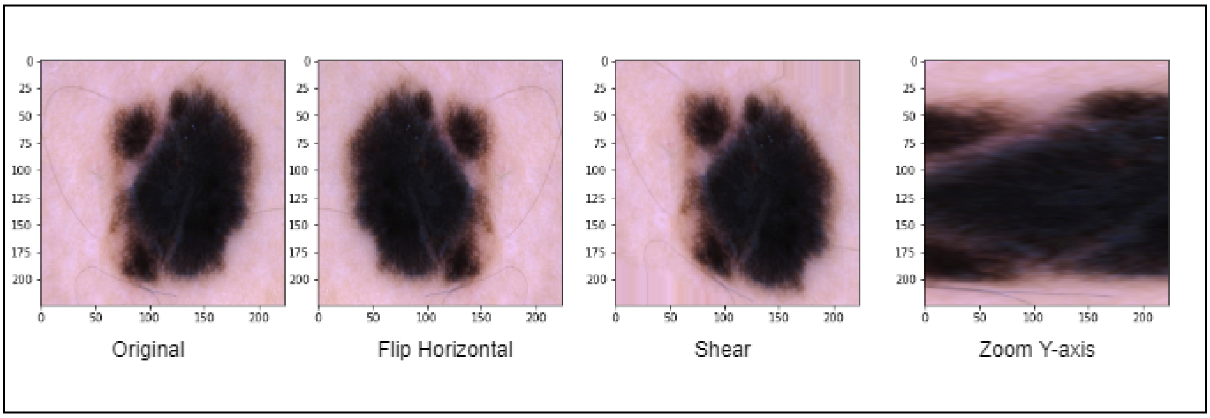

Figure 4: Data Augmentation

3.2.4 Streamlit Framework

The author has created a web-based interface for Deep Learning Models because the goal of the research was to create a classification model that can be used by medical researchers as well. Model selection comes next once all model training and testing has been completed. Based on factors such as accuracy, sensitivity, specificity, recall, and F1 score, the optimum deep learning architecture model is chosen. With the open-source Streamlit application, the author has created web applications for models in machine learning, deep learning, and data science. Streamlit code can be written in Python just like deep learning code. In the Implementation section, the Streamlit application technique is highlighted.

4 Design Specification

The research work's framework is shown in the figure 5. The process begins with getting data from Kaggle repositories. To avoid overfitting Data Augmentation techniques have been implemented as described in the above sections. As the data is already split into Train and Test folders, data augmentation had to be applied on these two individuals. The data that will be used for training will be passed on to five pre-trained models which are DenseNet121, DenseNet169, DenseNet201, VGG-16, and VGG-19. The author has chosen pre-trained models as they already have been trained on 14 million images and have desired weights. The model is trained on given inputs and the number of epochs is set. The performance of the model is tuned with certain hyper-parameter with multiple trial and error combinations. As the models built achieve satisfactory results, models are then compared and the best-chosen model's weights are selected in order to build an open-source web app for Machine Learning which predicts whether the given image of Skin is Benign or Malignant.

5 Implementation

5.0.1 System Environment

Since it was convenient to run two Python notebooks concurrently on the cloud, the author used the Kaggle Notebook environment. 16 Gigabytes of RAM and 13 Gigabytes

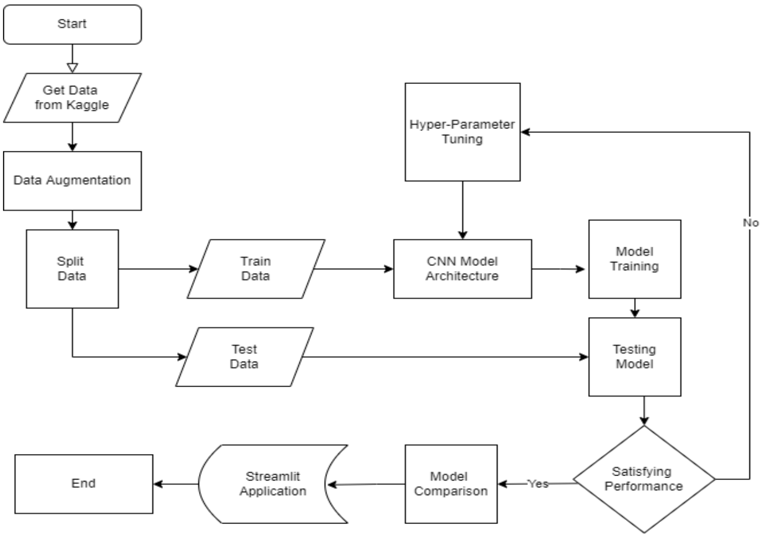

Figure 5: Design Specification

of GPU are the configurations offered.

The packages utilized were sci-kit-learn, matplotlib, and TensorFlow for deep learning model architectures, bar graphs, and evaluation metrics.

5.0.2 Model Design

Convolutional Neural Networks are frequently used in the field of medical image processing. Several specialists have made various attempts to create a model that can more precisely detect cancers. Author in this research has has developed a model that can classify Skin Images into Benign or Malignant. There are many Convolutional Neural Networks available, and among those, DenseNet and VGG have been implemented.

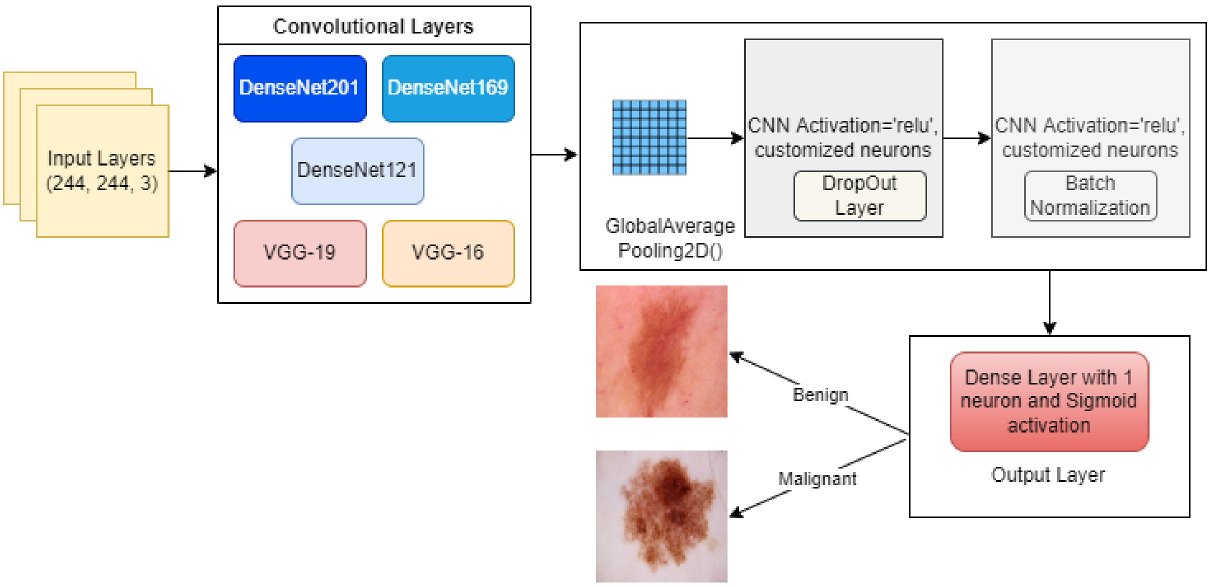

Figure 6: Model Architecture

All the models are built with different neuron combinations keeping 3 and 2 layers. Figure 6 shows how models will classify the binary outcome of Skin being Benign or Malignant. This research work has build DenseNet121, DenseNet169, DenseNet201, VGG-19 and VGG-16, keeping wide range of neurons in the layers such as (512, 256, 128), (512,256), (256,128, 64), (256, 128), (128, 64, 32), (128, 64). Total 30 models have been developed with these combinations.

5.0.3 DenseNet (DenseNet-121, DenseNet-169, DenseNet-201)

Densely connected Convolutional neural networks get their name from the way that each layer in the dense net architecture is closely related to every other layer. As mentioned above, this research has implemented DenseNet-121, DenseNet-169, and DenseNet-201 for classification. Every Layer in DenseNet takes information from its preceding one and builds a feature map that reduces networking. DenseNet is a perteained model. The author has set some modifications according to the problem statement which will be justified in Model architecture implementation.

5.0.4 VGG (VGG-16, VGG-19)

The VGG architecture of a deep neural convolution network accepts an image input of 224 x 224 pixels. It is made with a stride of 1 pixel and tiny 3 x 3 convolution layers spread across the 12 architecture. The author used the same layer setup for the VGG models that were used for DenseNet structures. In the section below, they are justified.

5.0.5 Model Configuration and Justification:

The top layer of DenseNet and VGG is removed by keeping the value as 'False' to turn off the output layer of 1000 neurons. 'Weights; are set to "imagenet', which means that model is trained on imagenet dataset that contains 14 million images of 1000 classes. All the layers are made non-trainable by keeping 'layer.trainable = False' which means freezing the layers so that in the model training process pre-trained model layers won't learn anything and just pass the information to the next layer.

The input shape is given as 224 * 224 * 3 which take the input X$_train, andY_trainthatispresentinNum$

Output from the Global Average Pooling layer will be passed to dense layer of either 512, 256, 128 (multiple numbers can be selected multiples of 32 or 64) neurons depending on the 12 DenseNet models as mentioned. Activation is set to 'Relu' because Relu learns the non-linear patterns better compared to other activation functions. Dropout layer is added to avoid overfitting of the model and is set to 0.25. This value should be between 0.1 and 0.5, but not more than 0.5, consequently, if kept above 0.5 the would have been no need of above dense layer. Furthermore, if author would have kept a higher dropout rate than 0.25, the training performance will decrease.

The dropout layer is placed after GlobalAveragePooling as, GlobalAveragPooling returns the feature vector and each value is present in the feature vector is important for detection. Batch Normalization layer is typically implemented after all other layers and prior to the output layer for the sole purpose of accelerating the training process and achieving good performance in fewer epochs. The output layer has one neuron with sigmoid activity as it is a binary classification system.

5.0.6 Hyper-Parameters and Epochs

Every hyper-parameter was also taken into account, and batch normalization and dropout layers were introduced to prevent overfitting. To increase the non-linearity of the cost function, the ReLu activation function has been utilized in between the convolution layers. Hyperparameters include the number of dense layers for each model (between 2 and 3), the number of neurons in each dense layer (multiples of 32), the number of epochs, the dropout layer, and batch normalization.

Epochs were set to 1000 with ModelCheckpoint and EarlyStopping callback functions, where patience was set to 5 and validation loss was monitored.

5.0.7 Implementation of Streamlit application

2 pages are built into this streamlit application one page is to describe the project, disease information, and for what purpose this streamlit app is utilized and the next page is to make predictions by uploading skin cancer images so that the app can be appropriate predictions by passing the uploaded image to a pre-trained model that shows the predicted class with predicted probability.

Once the image is uploaded to the web application and in the backend, it takes the image and converts it to a NumPy array of shapes 224, 224, and 3 of float type, and in the next step normalization is applied and further reshaped to 1, 224, 224, 3 because model accepts a 4-dimensional array. Finally, the model returns the predicted probability the value between 0 to 1 and based on the probability value if the value is greater than 0.5 then it will render as Malignant with probability value or else Benign

Streamlit Components that are used to build this web application are :

- **file_uploader:** This component provides upload functionality to users so that Users can upload images of different formats for further processing or predictions.
- **selectbox:** SelectBox component is used to select one value from different drop-down options in this application it is used to switch between two pages
- **header/ subheader/write:** functions like header, subheader, and write are used render text in user interface

6 Evaluation

It is important for physicians and those who build software to have an understanding of the suggested machine learning and Deep Learning models in order to improve care for patients. It is common practice to give multiple metrics when attempting to characterize the effectiveness of a model because there is no single statistic that can capture all of the desirable properties of a model. This section defines numerous metrics in the context of binary classification and calculates the most relevant metrics so that other researchers and physicians can easily apply the metrics mentioned in this work in their own study. This section also provides an overview of the work that was done. This section will place a significant amount of attention on the ratings for accuracy, sensitivity, and specificity out of all the metrics that are currently accessible.

- **Accuracy:** The accuracy of the evaluation dataset is the proportion of correctly classified samples. This statistic is frequently used in machine learning Deep Learning applications in medicine, although it may be misleading in situations where there are different proportions of the classes, as it is straightforward to attain high accuracy by classifying all samples into the dominant class. The range of the accuracy is [0, 1], with 1 denoting correctly predicting both positive and negative samples and 0 denoting neither.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

- **Sensitivity:**The recall is the ratio of correctly classified positive samples to all positive samples, commonly referred to as the sensitivity or True Positive Rate (TPR). The recall is limited to the range [0, 1], where 1 denotes correctly guessing the positive class and 0 denotes incorrectly estimating every sample in the positive class. Since it is preferred to miss as few good examples as possible, which corresponds to a high recall, this statistic is also thought to be among the most significant for medical investigations.

$$Sensitivity = Recall = \frac{TP}{TP + FN} \tag{2}$$

- **Specificity:** The specificity measures the proportion of negative samples that are correctly identified and is the negative class equivalent of sensitivity. It is determined as the ratio of samples that were correctly classified as negative to all samples that were negatively classified. The range of the specificity is [0, 1], with 1 denoting flawlessly detecting the negative class and 0 denoting inaccurately guessing all negative class samples.

$$Specificity = \frac{TN}{FP + TN} \tag{3}$$

The models that have been created to categorize skin lesions into benign and malignant conditions will be the main topic of this section. DenseNet and VGG, two deep learning approaches, have been used to build a total of 30 models by combining different layer arrangements with varied numbers of neurons as tuning factors. During the assessment phase, the process's losses and accuracy are calculated for each epoch for models. Plots of accuracy and loss are shown. The ratio of test data to predictions is used to calculate test accuracy. The confusion metrics for each model are calculated to get a real positive and real negative outcome. Metrics like Accuracy, Precision, F1 Score, AUC, Sensitivity, Specificity, etc. have been computed for each model. However, as previously mentioned, accuracy, sensitivity, and specificity are the most crucial variables to be taken into account in medical imaging.

6.1 Experiment 1: DenseNet Models

18 DenseNet models have been constructed, including 6 DenseNet-121 models, with each model having a different layer, and each layer consisting of a different set of neurons, such as (512, 256, 128), (512, 256), (256,128, 64), (256, 128), (128, 64, 32), and (128, 64). Similarly, 6 models for DenseNet-169 and 6 models for DenseNet-201 have been built.

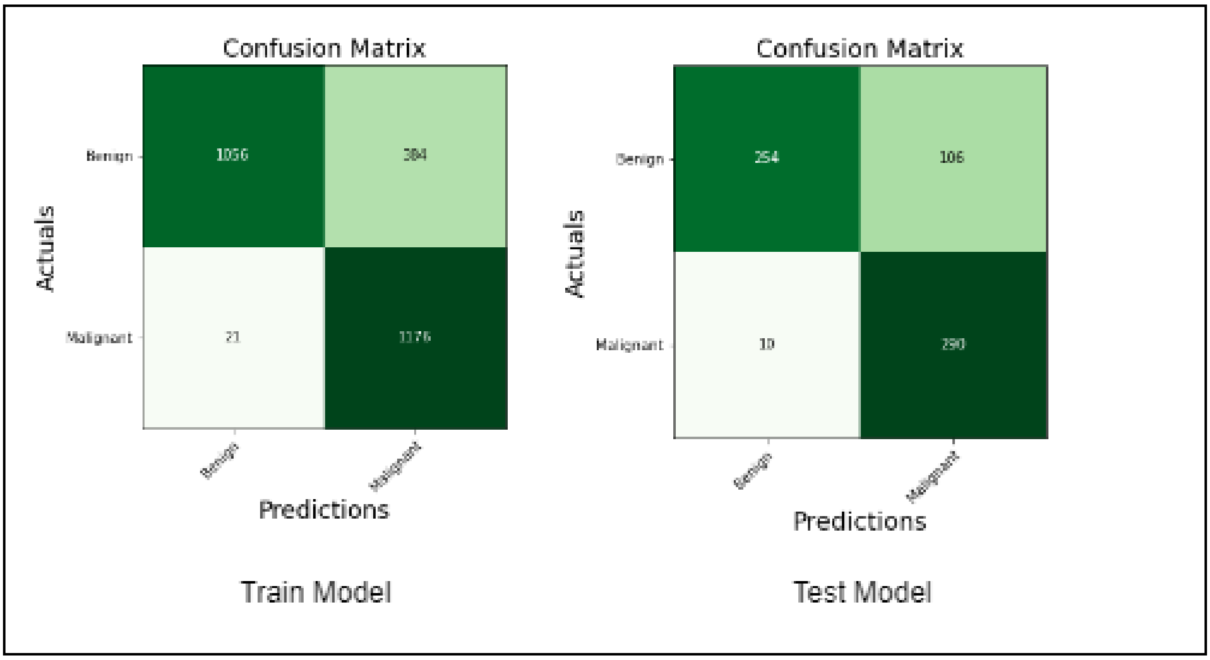

Figure 7: Configuration Matrix for DenseNet-121 (3 Layer)

In the subsequent section, each and every one of the DenseNet model architecture's performance metrics was shown.

The results showed that the DenseNet-121 model, which included three layers, 512, 256, and 128 neurons, was superior to the other 17 models. The testing model received a score of 84.64 for accuracy while achieving a sensitivity score of 96.66 and 70.55 for specificity. The Confusion matrix for both the test model and the trained model is shown in figure 7. Figure 8 provides a summary of all the metrics for DenseNet-121, which has three layers (512, 256, and 128 neurons).

	Data Partiton	Accuracy (%)	FAR (%)	FPR (%)	FNR (%)	Sensitivity (%)	Specifcity (%)	AUC (%)	Precision	Recall	F1 Score
0	Train Data	84.641638	15.358362	26.666667	1.754386	98.245614	73.333333	85.789474	75.384615	98.245614	85.310120
1	Test Data	82.424242	17.575758	29.444444	3.333333	96.666667	70.555556	83.611111	73.232323	96.666667	83.333333

Figure 8: Evaluation Metrics for DenseNet-121

Model losses and accuracy are estimated for each epoch throughout the assessment phase. Error and loss plots are shown in 9. A test's accuracy is a ratio of prediction and test data.

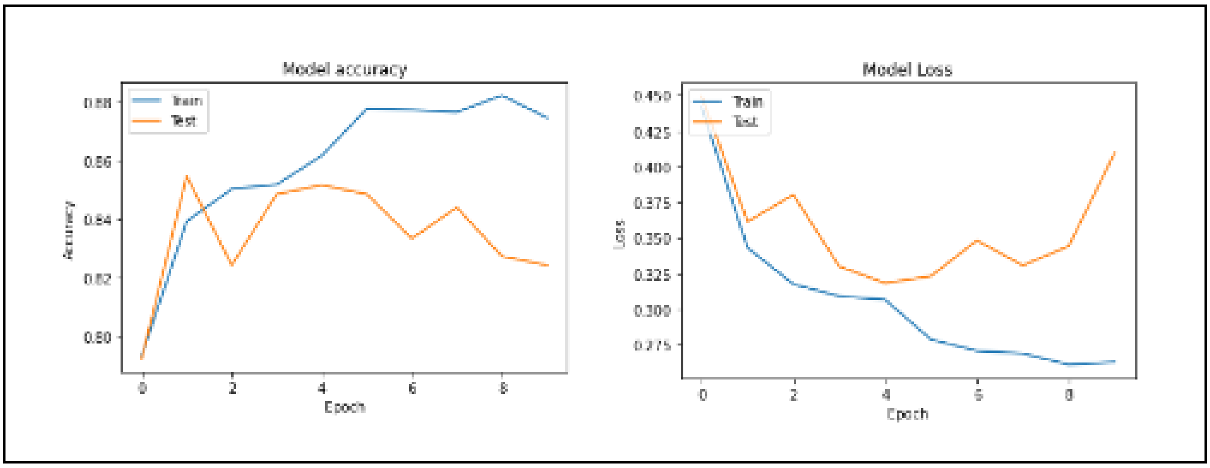

Figure 9: Graphs for DenseNet121 with 3 layer and (512, 256, 128) Neurons

6.2 Experiment 2: VGG Models

12 VGG models, including 6 VGG-16 models, have been created, each with a different layer and set of neurons, such as (512, 256, 128), (512, 256), (256,128, 64), (256, 128), (128, 64, 32), and (256, 128). (128, 64). VGG-19 has 6 models. The VGG-16 model, with 256, 128, and 64 neurons, outperformed the other 11 models. The model scored 79.21 for accuracy, 99.00 for sensitivity, and 60.55 for specificity.

Figure 10 shows the test and trained Confusion matrices.

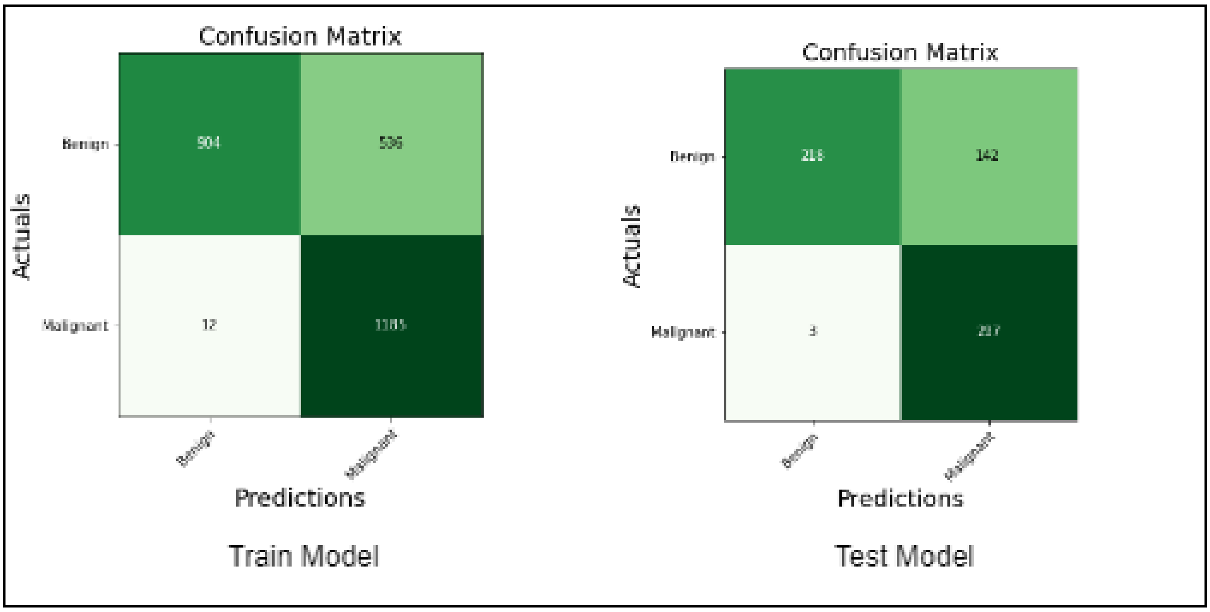

Figure 10: Configuration Matrix for VGG-16 (3 Layer)

Figure 11summarizes VGG-16's three-layer metrics (256, 128, 64 neurons). Throughout the assessment step, model losses and accuracy are estimated.

The figure 12 displays error and loss plots. An accuracy test compares prediction and test results.

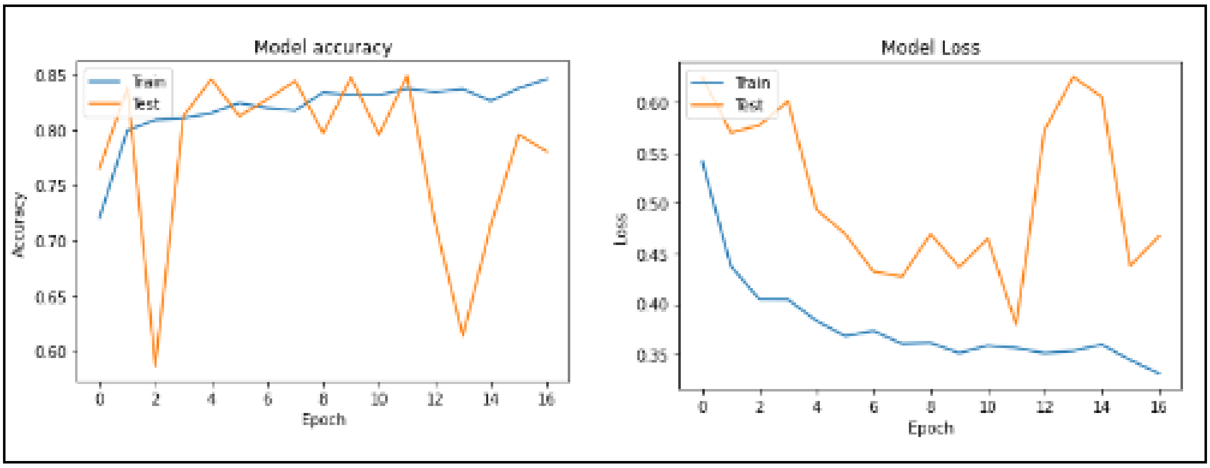

Figure 11: Graphs for VGG-16 with 3 layers and (256, 128, 64) Neurons

	Data Partiton	Accuracy (%)	FAR (%)	FPR (%)	FNR (%)	Sensitivity (%)	Specifcity (%)	AUC (%)	Precision	Recall	F1 Score
0	Train Data	94.463405	5.536595	5.625000	5.430242	94.569758	94.375000	94.472379	93.322341	94.569758	93.941909
1	Test Data	87.727273	12.272727	11.111111	13.666667	86.333333	88.888889	87.611111	86.622074	86.333333	86.477462

Figure 12: Evaluation Metrics for VGG-16

6.3 Evaluation Metrics for All 30 models

The author of this research has only focused on 3 evaluation metrics as discussed above, which are essential for Medical Imaging.

6.3.1 DenseNet-121

The evaluation metrics for the six models produced for DenseNet-121 are displayed in the following table 1.

Table 1: Evaluation metrics for 6 DenseNet-121 models

PRETRAINED MODEL NAME	# DENSE LAYERS	# NEURONS IN DENSE LAYERS	ACCURACY		Sensitivity		Specificity	
			TRAIN	TEST	TRAIN	TEST	TRAIN	TEST
DenseNet-121	2	128 , 64	88.24	84.69	82.28	80.33	93.19	88.33
DenseNet-121	2	256 , 128	90.67	85.60	86.54	83.66	94.09	87.22
DenseNet-121	2	512 , 256	88.66	83.03	96.99	93.33	81.73	74.44
DenseNet-121	3	128 , 64, 32	89.41	86.66	87.46	85.33	91.04	87.77
DenseNet-121	3	256, 128 , 64	86.31	82.57	95.32	92.33	78.81	74.44
DenseNet-121	3	512 , 256 , 128	84.64	82.42	98.24	96.66	73.33	70.55

6.3.2 DenseNet-169

The following table 2 shows the evaluation metrics for the six models created for DenseNet-169 where the model with 2 layers, 256 and 128 neurons performed better than others.

Table 2: Evaluation metrics for 6 DenseNet-169 models

PRETRAINED MODEL NAME	# DENSE LAYERS	# NEURONS IN DENSE LAYERS	ACCURACY		Sensitivity		Specificity	
			TRAIN	TEST	TRAIN	TEST	TRAIN	TEST
DenseNet-169	2	128 , 64	89.15	86.66	86.04	83.66	91.73	89.16
DenseNet-169	2	256 , 128	90.70	87.12	89.22	85.00	91.94	88.88
DenseNet-169	2	512 , 256	91.46	86.96	85.46	79.00	96.45	93.61
DenseNet-169	3	128 , 64, 32	89.87	86.51	88.63	84.66	90.90	88.05
DenseNet-169	3	256, 128 , 64	88.50	85.90	79.19	76.33	96.25	93.88
DenseNet-169	3	512 , 256 , 128	83.57	81.96	67.25	66.00	97.15	95.27

6.3.3 DenseNet-201

The evaluation metrics for the six models developed for DenseNet-201 are shown in the table 3 below, with the best model having two layers and 512 and 256 neurons.

Table 3: Evaluation metrics for 6 DenseNet-201 models

PRETRAINED MODEL NAME	# DENSE LAYERS	# NEURONS IN DENSE LAYERS	ACCURACY		Sensitivity		Specificity	
			TRAIN	TEST	TRAIN	TEST	TRAIN	TEST
DenseNet-201	2	128 , 64	89.07	84.39	80.45	74.33	96.25	92.77
DenseNet-201	2	256 , 128	88.62	85.30	80.36	75.00	95.48	93.88
DenseNet-201	2	512 , 256	92.64	86.06	93.06	86.66	92.29	85.55
DenseNet-201	3	128 , 64, 32	89.19	84.54	82.20	75.33	95.00	92.22
DenseNet-201	3	256, 128 , 64	87.90	83.78	84.71	79.00	90.55	87.77
DenseNet-201	3	512 , 256 , 128	90.44	85.00	87.55	81.33	92.84	88.05

6.3.4 VGG-16

The evaluation metrics for the six VGG-16 models are shown in the table below 4, with the best model having three layers with 256, 128, and 64 neurons.

Table 4: Evaluation metrics for 6 VGG-16 models

PRETRAINED MODEL NAME	# DENSE LAYERS	# NEURONS IN DENSE LAYERS	ACCURACY		Sensitivity		Specificity	
			TRAIN	TEST	TRAIN	TEST	TRAIN	TEST
VGG-16	2	128 , 64	81.72	81.06	66.66	67.00	94.23	92.77
VGG-16	2	256, 128	85.70	84.84	93.73	92.66	79.02	78.33
VGG-16	2	512 , 256	83.95	84.24	96.07	95.33	73.88	75.00
VGG-16	3	128 , 64, 32	85.47	84.09	85.46	84.66	85.48	83.61
VGG-16	3	256, 128 , 64	79.21	78.03	98.99	99.00	62.77	60.55
VGG-16	3	512 , 256 , 128	85.55	85.45	86.71	89.00	84.58	82.50

6.3.5 VGG-19

The evaluation metrics for the six VGG-19 models are shown in the table 5 below, with the best model having three layers with 512, 256, and 128 neurons.

Table 5: Evaluation metrics for 6 VGG-19 models

PRETRAINED MODEL NAME	# DENSE LAYERS	# NEURONS IN DENSE LAYERS	ACCURACY		Sensitivity		Specificity	
			TRAIN	TEST	TRAIN	TEST	TRAIN	TEST
VGG-19	2	128 , 64	81.83	80.90	68.25	68.66	93.12	91.11
VGG-19	2	256 , 128	82.32	81.36	91.14	91.33	75.00	73.05
VGG-19	2	512 , 256	68.79	68.48	32.99	33.00	98.54	98.05
VGG-19	3	128 , 64, 32	77.51	77.27	54.30	55.00	96.80	95.83
VGG-19	3	256, 128 , 64	81.75	81.66	70.92	73.66	90.76	88.33
VGG-19	3	512 , 256 , 128	78.11	75.75	98.74	98.66	60.97	56.66

6.4 Discussion and Result interpretation

The goal of this research was to improve the accuracy, sensitivity, and specificity of categorizing benign and malignant skin images. The environment, which makes use of Kaggle's Notebook, aids in speeding up processing while providing results that are remarkably precise. All of the models' average accuracy is good, but critical parameters like sensitivity and specificity are given more weight. Not only model's great accuracy has not led to it being regarded as the best model; rather, a balance between sensitivity and specificity has been upheld along with the Accuracy score.

In terms of the top models, DenseNet-121, which has three layers with 512, 256, and 128 neurons in each, scored 96.99 percent for sensitivity and 70.55 percent for specificity. Its accuracy was 82.42 percent. In other words, DenseNet-121 with 3 layers can accurately identify a patient having skin cancer 96.66% of the time and an illness 3.44% of the time. In terms of specificity, the model can reliably identify patients without skin cancer 70.55% of the time.

Out of 30 deep-learning models, 5 were chosen as the best performers, as indicated in the table 6. Although VGG-19 has the greatest sensitivity score, DenseNet-121 can be considered the top-performing model in this research effort when accuracy and specificity are taken into account.

Table 6: Top Performing Models and their Evaluation Metrics

PRETRAINED MODEL NAME	# DENSE LAYERS	# NEURONS IN DENSE LAYERS	ACCURACY		Sensitivity		Specificity	
			TRAIN	TEST	TRAIN	TEST	TRAIN	TEST
DenseNet-121	3	512 , 256 , 128	84.64	82.42	98.24	96.66	73.33	70.55
DenseNet-169	2	256 , 128	90.70	87.12	89.22	85.00	91.94	88.88
DenseNet-201	2	512 , 256	92.64	86.06	93.06	86.66	92.29	85.55
VGG-16	3	256, 128 , 64	79.21	78.03	98.99	99.00	62.77	60.55
VGG-19	3	512 , 256 , 128	78.11	75.75	98.74	98.66	60.97	56.66

As a result, DenseNet-121 can be implemented when developing web-based applications so that the end user can classify Skin Lesions using a single image.

7 Conclusion and Future Work

Five models of transfer learning To create 30 models, architectures were tuned in various levels and parameters. The dataset used to test these models was obtained from Kaggle repositories and was divided into train and test data. With DenseNet-201 accounting for the highest accuracy, the top 5 models have all achieved greater than 80% accuracy. DenseNet-121 with three layers outperformed all other models in this study because accuracy, sensitivity, and specificity were all given equal weight. Accuracy was 84.64%, sensitivity was 96.66, and specificity was 70.55. In comparison to earlier models reported in literature surveys, the hyper-parameters employed in various layers and configurations and the parameters specified for epochs have produced excellent results. A web-based application for medical stakeholders has been developed using the best-performing model and deployed to Python's Streamlit Framework.

For future work, the web-based application could be built with multiple models that could retrieve the results for each individual model while comparing them with each other. Similar to how skin cancer is categorized, other medical imaging fields can also be taken into account and built upon to increase their Accuracy, Sensitivity, and Specificity scores.

8 Acknowledgement

I want to recognize and express my gratitude to Professor Noel Cosgrave for his significant contribution to my academic performance. Without him, I would not have been able to achieve this level of success. I am really appreciative of his assistance, direction, and insightful criticism during the study process.

References

Ali, M. S., Miah, M. S., Haque, J., Rahman, M. M. and Islam, M. K. (2021). An enhanced technique of skin cancer classification using deep convolutional neural network with transfer learning models, *Machine Learning with Applications* **5**: 100036.

Feng, S., Yang, Z., Huang, M. and Wu, Y. (2022). Classification of crop disasters based on densenet, *2022 5th International Conference on Pattern Recognition and Artificial Intelligence (PRAI)*, pp. 252–256.

Hameed, N., Shabut, A. M. and Hossain, M. A. (2018). Multi-class skin diseases classification using deep convolutional neural network and support vector machine, *2018 12th International Conference on Software, Knowledge, Information Management Applications (SKIMA)*, pp. 1–7.

Hamwi, W. A. and Almustafa, M. M. (2022). Development and integration of vgg and dense transfer-learning systems supported with diverse lung images for discovery of the coronavirus identity, *Informatics in Medicine Unlocked* **32**: 101004.

Jahan, N., Anower, M. S. and Hassan, R. (2021). Automated diagnosis of pneumonia from classification of chest x-ray im ages using efficientnet, *2021 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD)*, pp. 235–239.

Jain, V., Kavitha, H. and Mohana Kumar, S. (2022). Credit card fraud detection web application using streamlit and machine learning, *2022 IEEE International Conference on Data Science and Information System (ICDSIS)*, pp. 1–5.

Kaur, I., Kumar, Y. and Sandhu, A. K. (2021). A comprehensive survey of ai, blockchain technology and big data applications in medical field and global health, *2021 International Conference on Technological Advancements and Innovations (ICTAI)*, pp. 593–598.

Kumar, A. A., Krishna Reddy, M. V., Sanjay, V. M., Kothuru, V., Shah, K. P. and Kalucha, A. (2022). Comparative analysis of skin cancer prediction using neural networks and transfer learning, *2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS)*, Vol. 1, pp. 1518–1521.

Labde, S. and Vanjari, N. (2022). Prediction of skin cancer using cnn, *2022 3rd International Conference for Emerging Technology (INCET)*, pp. 1–4.

Li, X., Zhang, J., Xue, Y. and Qiu, L. (2022). Classification of hops image based on resnet-convlstm and research of intelligent liquor picking system, *Measurement* **194**: 110955.

Majib, M. S., Rahman, M. M., Sazzad, T. M. S., Khan, N. I. and Dey, S. K. (2021). Vgg-scnet: A vgg net-based deep learning framework for brain tumor detection on mri images, *IEEE Access* **9**: 116942–116952.

Ramachandro, M., Daniya, T. and Saritha, B. (2021). Skin cancer detection using machine learning algorithms, *2021 Innovations in Power and Advanced Computing Technologies (i-PACT)*, pp. 1–7.

Sarwinda, D., Paradisa, R. H., Bustamam, A. and Anggia, P. (2021). Deep learning in image classification using residual network (resnet) variants for detection of colorectal cancer, *Procedia Computer Science* **179**: 423–431. 5th International Conference on Computer Science and Computational Intelligence 2020.

Sharma, G. and Chadha, R. (2022). Skin cancer and oral cancer detection using deep learning technique, *2022 IEEE International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE)*, pp. 1–5.

Shehab, M., Abualigah, L., Shambour, Q., Abu-Hashem, M. A., Shambour, M. K. Y., Alsalibi, A. I. and Gandomi, A. H. (2022). Machine learning in medical applications: A review of state-of-the-art methods, *Computers in Biology and Medicine* **145**: 105458.

Singh, P., Kumar, M. and Bhatia, A. (2022). A comparative analysis of deep learning algorithms for skin cancer detection, *2022 6th International Conference on Intelligent Computing and Control Systems (ICICCS)*, pp. 1160–1166.

Wang, Q., Yang, B., Liu, W. and Chen, G. (2021). X-ray images detection of covid-19 based on deepwise separable densenet, *2021 IEEE 6th International Conference on Signal and Image Processing (ICSIP)*, pp. 294–298.

Yalamarthi, R. N. S. V., Shaik, S., Singh, D. and Rakhra, M. (2022). Real-time face mask detection using streamlit,tensorflow, keras and open-cv, *2022 IEEE International Conference on Data Science and Information System (ICDSIS)*, pp. 1–5.

Zhang, Y., Jin, Z., Liu, F., Zhu, W., Mu, W. and Wang, W. (2020). Imagedc: Image data cleaning framework based on deep learning, *2020 IEEE International Conference on Artificial Intelligence and Information Systems (ICAIIS)*, pp. 748–752.