An Enhanced CNN Model For Classifying Skin Cancer

by

Kazi MD Minhajul Haider
18101708
Mondira Dhar
21241070
Fahima Akter
19101642
Sadia Islam
18301232
Syed Ragib Shariar
18101571

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

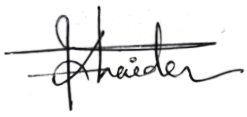
Department of Computer Science and Engineering
Brac University
January 2022

Declaration

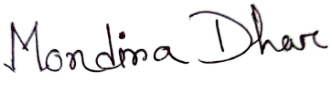
It is hereby declared that

1. The thesis submitted is our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

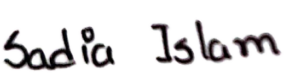


Kazi MD Minhajul Haider
18101708

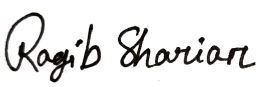


Mondira Dhar
21241070

Fahima Akter
19101642



Sadia Islam
18301232



Syed Ragib Shahriar
18101571

Approval

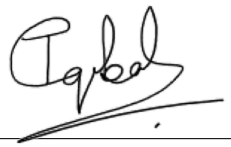
The thesis titled "An Enhanced CNN Model For Classifying Skin Cancer" submitted by

1. Kazi MD Minhajul Haider (18101708)
2. Mondira Dhar (21241070)
3. Fahima Akter (19101642)
4. Sadia Islam (18301232)
5. Syed Ragib Shariar (18101571)

Of Fall, 2021 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on January 16, 2022.

Examining Committee:

Supervisor:
(Member)



Dr. Muhammad Iqbal Hossain
Assistant Professor
Department of Computer Science and Engineering
Brac University

Co-Supervisor:
(Member)

Moin Mostakim
Lecturer
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Dr Md Golam Rabiul Alam
Associate Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi
Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

Unrepaired deoxyribonucleic acid in skin cells causes skin cancer by generating genetic abnormalities or mutations, rising day by day. Detecting and diagnosing skin cancer in its early stages is expensive and challenging, giving superior treatment options. Given the severity of these issues, researchers have generated a set of early classification techniques for skin cancer. Skin cancer is diagnosed and segregated from melanoma by looking at the symmetry, color, size, shape, and other features of lesions. While there are various computerized approaches for classifying skin lesions, convolutional neural networks (CNNs) have been demonstrated to exceed standard practices. Moreover, CNNs are a type of deep learning that has been prominent in various fields, including medical imaging. Multiple machine learning libraries have been used in this paper. Also, we have used five pre-trained models such as Inception V3, VGG-19, VGG-16, Efficient Net B7, ResNet 50 models and presented our proposed model for skin cancer classification using the HAM10000 dataset, which is an enormous skin cancer dataset. Following that, each competent model's image detection categorization accuracy is evaluated by comparing and assessing. This research reports a maximum accuracy of 85.25% for Inception V3 models within five pre-trained models and maximum accuracy of 90.55% for our proposed model. In terms of image detection, our experimental configuration shows that our proposed model can attain the best classification accuracy rather than the other five pre-trained models. Our findings are helpful in providing a comprehensive comparison and analysis of many neural networks in the categorization of skins cancer.

Keywords: Skin Cancer; CNN; Deep Learning; Medical Imaging; Accuracy

Acknowledgement

First of all, all praise to the Great Almighty, as we could finish our thesis without any serious impediment. Second of all, to our supervisor, Dr. Mohammad Iqbal Hossain, and co-supervisor, Mr. Moin Mostakim, who consistently supported andguided us throughout this project. Finally, to our parents, whose perennialsupport and encouragement played a pivotal role in completing this thesis.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

According to the World Health Organization (WHO), cancer is one of the leading causes of death[10]. Again, according to statistics, more than two people in the United States die from skin cancer every hour. Skin cancer is classified into two types: melanoma and non-melanoma. Every year, 132,000 melanoma skin cancers and around 3 million non-melanoma skin cancers are diagnosed worldwide[10]. Melanoma is a kind of skin cancer that is deadly, rare, and lethal. According to the American Cancer Society, it is estimated to account for just 1% of all cases, although it is linked to a greater survival rate. Melanoma is a type of cancer that occurs in cells called melanocytes. Moreover, It begins when normal melanocytes grow out of control, resulting in a malignant tumor. It can impact any portion of the human body over time. Melanoma skin cancer manifests itself in a range of forms, including nodular melanoma, superficial spreading melanoma, acral lentiginous, and lentigo maligna.

Non-melanoma cancers, such as squamous cell carcinoma (SCC), basal cell carcinoma (BCC), and sebaceous gland carcinoma, account for the vast majority of cancer cases (SGC). BCC, SGC, and SCC are generated in the epidermis' middle and higher layers, respectively. These cancer cells have a low proclivity for spreading to other places of the body. Non-melanoma cancers are less difficult to treat than melanoma malignancies[20].

UV rays, heredity, an unhealthy lifestyle, and smoking are all factors of skin cancer. UV radiation has been identified as the leading cause of the majority of skin cancer incidences. Including the continuous breakdown of the ozone layers, quite disastrous UV radiations penetrate the earth's surface and elevate the number of skin cancer cases.

However, if cancer is discovered early, it may be treated with a simple incision, but later stage detection is associated with a higher risk of mortality. According to the statistics, if the diagnosis of skin cancer is at an early stage, the predicted 5-years survival rate is greater than 95%. On the other hand, if the diagnosis is comparatively late the survival rate is going to less than 20%[7].

Dermatoscopy helps to diagnose skin cancer early which refers to a non-invasive procedure for examining pigmented skin lesions at the surface level. As skin cancer has various forms, it illustrates that early detection is often challenging even for expert dermatologists. For example, It may have seemed like a form of swelling, moles which are sometimes difficult to detect with the naked eye. As a result, it is tough to understand if cancer is benign or malignant.

For highly non-linear classification, hospitals are moving forward to use Computer-Aided Diagnosis (CAD) systems. From an image processing perspective, CAD examines different characteristics such as texture, color, and form to identify whether a dermoscopic picture is cancerous or benign. Furthermore, the rise of powerful image classification algorithms gives the patients the appropriate path of treatment.

Furthermore, this study focuses on the provision of a complete, structured literature overview of conventional deep learning techniques e.g. Convolution neural network (CNN) followed by some general procedures such as image preprocessing, segmentation, identifying and categorizing the required features.

1.2 Motivation

As the medical field evolves, the perception of computer-assisted clinical decision support has become a huge concern in research to improve the quality of medical and healthcare decision-making. By easing diagnosis and remedial decisions, AI has the potential to boost individualized care.As a result, the focus of this research is to figure out how adding AI technologies might speed up the detection of skin cancer and design a system that relies on data accuracy for better decision-making.The objective is to provide a system that combines image processing ideas with Convolutional Neural Network models to help healthcare professionals make more accurate and effective assessments.

This study's goals are as follows:

1. To get a deep understanding of AI applications and how they might be applied to our chosen field.

2. To compare and evaluate the performance of conventional and transfer learning approaches to Convolutional Neural Networks.

3. To explore and assess the most effective model for our research and to assess the CNN model that has been proposed.

1.3 Aims and Objectives

This research aims to develop a CNN classification model for classifying skin cancers at an early stage. We explored different predefined CNN models to find out their accuracy on our dataset. Later we will compare our proposed model with the existing pre-defined model.

1. To deeply understand CNN classification
2. To deeply understand skin cancer classification techniques
3. To develop a CNN classification model for the classification of skin cancer based on CNN.
4. To evaluate the proposed CNN model
5. To offer recommendations on improving the model accuracy.

1.4 Research Problem

The prevalence of skin cancer is the most frequent chronic disease among the total prevalence of other types of cancers. The most severe kind of skin cancer is melanoma. If the cancer is detected early, the treatment is going to be 99% effective. However, if skin cancer is not diagnosed and treated on time, the rate of efficacy will be extremely low, and the survival rate will drop drastically. Apparently, physicians diagnose cancer by determining if the level of the lesions is potentially benign or malignant. Physicians take much longer to diagnose them because skin cancer and benign tumor lesions have similar shapes. Some of the symptoms included a large brownish spot with darker speckles, size or feel or that bleeds, a mole that changes in color, and painful lesions that itches or burns. Moreover, a tiny lesion with an uneven border with areas that seem red, pink, white, blue, or blue-black.

Currently, the Computer-aided diagnosis system known as CAD system is being used widely. Because this method is good at screening and detecting melanoma at an early stage. It takes Dermoscopic images as an input and performs image processing. Then determine whether the cancer is positive or negative, if positive then malignant or benign[10]. However, this CAD system is not capable of detecting the actual type of skin cancer. At this point, there is a need for a fast and accurate classification system through which we can determine the actual classification of skin cancer. If medical staff are able to find out the correct classification, then the chances of getting proper treatment and duration will increase. So, the mortality rate will decrease.

In this classification field using Desmoscopic images, Convolution Neural Networks (CNN) appeared as a powerful solution that focused on the diagnosis of skin cancers. There are some pre-defined transfer learning methods that are very popular, known as VGG16, VGG19, AlexNet, ResNet, googleNet. VGG methods have some cons, like too many weight parameters, which makes the model heavy, and the inference time is high. However, VGG is much deeper. Again, Alex net is not very deep, and it faces problems to scan all the features in a massive dataset. Moreover, This model was built for a certain use case.

From this point of view, we felt there is a need for an enhanced CNN-based model which will perform well in a large dataset, have less execution time and be easy to implement. Besides, we also perform other existing popular models on HAM10000

to figure out which model performs well in this imbalance dataset which will help medical staff to pick the correct model for making the classification.

1.5 Contribution and impact

The goal of our study is to prove that our multiple technique of comparing various transfer learning models and the proposed model is efficient in finding the optimum model that works best for classifying skin cancer. We were also able to assess the model more precisely on various sizes by comparing it with various parameters. We implemented the models Efficient net, VGG-16, VGG-19, Inception v3, and our proposed model to perform multi-label classification. In both processes, we used both pre-trained and arbitrarily initialized weights to comprehend the models' performance.

We believe the medical sector requires a more correct model to make the classification, an advanced CNN-based model that will perform well in huge datasets, take less time to execute, and be simple to deploy. Despite the fact that there have been several studies on skin cancer detection, the most of them have concentrated on determining the accuracy rate of a few models. Moreover, we have used four other transfer learning models, as well as our own proposed model. To conclude, our research shows that, of all the models evaluated, our proposed model performed the best in our research which is classifying the appropriate data properly.

1.6 Scopes and Limitations

Though we managed to get very good results from our model, there were some limitations to our model. For instance, it can only classify seven types of skin cancer. There are several other types of lesions that are beyond our model's scope. Again, in order to detect cancer, it needs to be a fully formed lesion. In some cases, it might be too late for the patient. As it is a matter of medical emergency, it would have been better if our model had more than 95% accuracy in classifying cancer types. Our dataset also had an uneven distribution of data. That is why we had to use augmentation. In terms of the augmented classes, the model performs slightly lesser than the other classes. If the dataset had a balanced distribution, the model would have performed better for all the classes.

1.7 Documentation Outline

This section provides an outline of the topics covered in each chapter of this thesis paper. The remainder of the article is organized in the following manner after describing the purpose for this study and what we desire to achieve and plan to accomplish in this chapter:

1. **Chapter 1** includes the introduction Here, we talk about the background, motivation, objectives, research problem, contribution and impact also, scopes, and limitations.

2. **Chapter 2** includes literature review and related works
3. **Chapter 3** explain the methodology
4. **Chapter 4** covers the pre-trained transfer learning models
5. **Chapter 5** covers the proposed models
6. **Chapter 6** discusses model complications
7. **Chapter 7** evaluates the experimental result and performance analysis
8. **Chapter 8** outlines the accuracy comparison and related works
9. **Chapter 9** concludes this thesis with subsequent plan

Chapter 2

Literature Review and Related Works

In this paper[21] the authors train a multi layer perceptron, a custom CNN model and VGG-16 to classify skin cancer. Among these three VGG-16 performed the best. VGG-16 is a pre-trained model on the imagenet dataset. They have made discoveries such as the time computation for CNN is longer than that of MLP and training for CNN can be done offline. Various tables and graphs are shown which clearly show that the VGG-16 model gives the best performance for the classification for skin diseases. The dataset only had seven types of skin diseases, however, there are several other skin diseases which are not addressed. The paper was conducted only on one dataset, results may vary if they were implemented on other dataset. Their main objective was to compare different methods on the same dataset, HAM10000. We have studied other papers, one being "Automatically Early Detection of Skin Cancer" by Ho Tak Lau and Adel Al-Jumaily, but they failed to meet our requirements as they used only one algorithm.Different algorithms were used on to the chosen dataset to get a bigger perspective.
Every year thousands of unnecessary biopsies are done to detect skin cancer. Skin cancer always starts a skin lesion so detecting them earlier would make dermatologists' lives easier. The authors used different CNN models using transfer learning techniques. Among them, ResNet50 showed the best output. The accuracy achieved from ResNet50 was at 90%. The main challenge the authors faced was data imbalance in the dataset. To balance the dataset and to avoid overfitting they used data augmentation, max pooling and neuron dropout. They also used Mobilenet, Xception and InceptionV3. But considering everything, ResNet50 was the best model. As their future plan they suggested that using Segmentation on the images and using them as input to the model might improve the performance of the model[13].

One of the papers[7] claimed that using a variety of pre-trained models and extracting features from several layers can result in high accuracy. Their approach utilized three deep learning models known as ResNet-18, AlexNet, and VGG16. They are pre-trained on ImageNet as efficient feature extractors and SVM, with classifiers trained on a sampling of pictures from the ISIC repository. They combined the SVM outputs to achieve optimal distinction among three lesion classes in the final stage.

In [17], the author proposed an improved way of classifying skin cancer using DCNN with transfer learning which provides more accuracy even if the patients are in the early stages of the disease. Their proposed DCNN model scores much better results than other DL models with a massive dataset. Moreover, the completion time of their proposed DNCC model is deficient compared to other pre-trained models like AlexNet, VGG-16. Finally, they compared their proposed DNCC with the Transfer Learning model and showed that their accuracy is better.

In this paper[10], the author has proposed a deep CNN method and used the HAM10000 dataset. He pre-processed the dataset and then implemented it. Finally, he trained and evaluated it. Moreover, for comparison, he used two pre-trained transfer learning methods, which are VGG16 and VGG19. He used overall average accuracy and loss metrics to compare the proposed DCNN and VGG16 and VGG19. After reaching the end, we can say that DCNN is superior to VGG16 and VGG19.

In this paper[2], the author suggested a deep learning method which is a multi-scale convolutional neural network to classify skin lesions. He used a pre-trained ImageNet dataset which has been fine-tuned for skin lesion categorization using two distinct picture scales as input and used the Inception-v3 deep neural network on their approach. In addition, the two scales are equivalent to a rough scale that identifies the lesion's general context and structural features. In contrast, the image at the more precise scale exposes textural details and various low-level characteristics of the lesion that are essential for separating the classes of lesions.

In this paper[19], the authors present detailed and methodical literature review of traditional deep learning approaches for skin cancer detection, including artificial neural networks (ANN), convolutional neural networks (CNN), Kohonen self-organizing neural networks (KNN), and generative adversarial neural networks (GAN).To write their paper, they have selected different papers from reputed journals and conferences based on some crucial criteria and among them, they have taken 51 papers and analyzed those from different aspects. Again, this paper creates a comparative vision of different approaches to neural network techniques via a table. To conclude their paper, they have noticed that as CNN is more directly tied to computer vision than other types of neural networks, it performs better when classifying image data. After reviewing their papers, they have suggested that Image acquisition will be automated and speed up with an autonomous full-body imaging process.

The main objective of the paper[20] is to convey a system that uses CNN to identify and classify skin cancer into distinct groups, and then uses the transfer learning approach to improve accuracy. Deep learning models pre-trained on the ImageNet dataset were utilized for Transfer learning. By appending additional layers and freezing portions of the prior levels, the HAM10000 dataset is then utilized to further train these pre-trained models. To compare the outcomes, they used different learning algorithms such as XGBoost, SVM, and Random Forest Algorithms to

complete the classification task in the HAM10000 dataset. They also discovered that in the HAM10000 dataset, learning methods such as Random Forest, XGBoost, and SVMs are ineffective for classification tasks.

In this paper[14], the authors implied a custom CNN based method to predict skin cancer from pictures. Their proposed model has a tremendous classification achievement comparable with two very prominent CNN architectures such as VGG-16 and VGG-19. First of all they have categorized skin cancer into nine varieties which are the vastly classified categories of skin cancers. Additionally they have used a dataset from 'kaggle.com' containing 25,780 pictures of benign and malignant cells. Also each picture was categorized to the definition brought from ISIC. The precision of their proposed custom CNN model is 79.45%, VGG-16 is 69.57% and VGG-19 is 71.19% . Finally, the detection rate which is suggested by the authors is more acceptable [13] than the VGG-16 model and the VGG-19 model.

In this paper[11], an automated system is proposed by the authors to identify the situations of different types of skin cancers from digital image processing. However the CNN model is adopted in this paper[11] containing 3 hidden layers which uses 3×3 filter sizes with 16, 32, and 64 channel outcomes in cycle and has a completely connected layer and softmax activation. They have conducted their optimization utilizing Adam, SGD, RMSprop, and Nadam optimizer on their proposed model and eventually based on their testing the authors found that the Adam optimizer gives the best interpretation of skin cancer lesions with 99% of accurateness from the dataset with the number of precision, recall and F1- is nearly 1.

Chapter 3

Methodology

In this chapter, we are going to describe the methodology for this thesis work. We started with the workflow with the collection of suitable dataset and preparation of the dataset by applying pre-processing techniques. The work flow consist of propose a model and compare the performance of the model with five pre trained transfer learning models such as inception v3, vgg16, vgg19, efficient net b7, and resNet 50. The workflow ends with evaluation of the performance of the models on the dataset. The methodology is sequentially outlined in the following steps :

step1. Collect the dataset
step2. Dataset pre-processing
step3. Pre-trained CNN models
step4. Proposed a suitable CNN model
step5. Evaluate the CNN model's performance

3.1 Workflow of the Methodology

Figure 3.1: Workflow of the Methodology

3.2 Observation of Dataset and preprocessing

3.2.1 Dataset collection and description

The dataset we are using for our thesis is the HAM10000 dataset. The dataset contains a CSV file containing the metadata of the image and two folders of lesion images. For our experiment, we used Human Against Machine (HAM10000) with 10015 training images dataset. It's a collection of dermatoscopic pictures from various populations that were obtained and retained through multiple techniques. The

HAM10000 training set's 10015 dermatoscopic images were gathered over a 20-year period from two separate locations: the Department of Dermatology at the Medical University of Vienna, Austria, and Cliff Rosendahl's skin cancer practice in Queensland, Australia. Dermatoscopy is a frequently used diagnostic procedure that, when compared to unaided eye inspection, enhances the identification of benign and malignant pigmented skin lesions. Dermatoscopic images are an excellent source for training artificial neural networks to automatically identify pigmented skin lesions. It is currently the leading source for dermatoscopic image analysis research due to permissive licensing, well-structured availability, and huge size. All important diagnostic categories of skin cancer had been added in this collection.There are a total of seven classes in the dataset. The columns of the dataset are described below:

Column name	**Description**
Index	The index number of the row
lesion_id	Identification number of the lesion
image_id	Unique identification number of the image
dx	Cancer type
dx_type	How the lesion was detected
Age	Age of the patient
Sex	Gender of the patient
Localization	Lesions are placed on the body

Table 3.1: Dataset description table

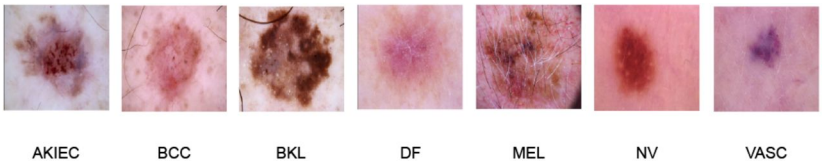


Figure 3.2: Types of skin cancer

More than 50% of the lesions are pathologically confirmed[7]. Seven classes of the dataset are given below:

1. **Bowen's disease(Akiec):** Actinic keratoses (Solar Keratoses) and intraepithelial carcinoma (Bowen's disease) are common noninvasive squamous cell carcinoma variants that can be treated locally without surgery.

2. **Basal cell carcinoma(Bcc):** Basal cell carcinoma is an epithelial skin cancer that seldom metastasizes but progresses rapidly if left untreated.

3. **Benign keratosis-like lesions(Bkl):** "Benign keratoses" include seborrheic keratoses (sometimes called "senile warts"), solar lentigo (a flat form of seborrheic keratosis), and lichen-planus-like keratoses (LPLK).

4. **Dermatofibroma(Df):** Dermatofibroma is a benign skin lesion that can either be a benign growth or an inflammatory reaction to slight trauma.

5. **Melanocytic nevi(Nv):** Melanocytic nevi are benign melanocytic neoplasms that can take many different shapes. The variances may differ considerably from a dermatoscopic perspective. They are often symmetrical in color and structural distribution, unlike melanoma.

6. **Vascular lesions(Vasc):** Cherry angiomas, angiokeratomas, and pyogenic granulomas are among the vascular skin lesions observed in the dataset. This category also includes hemorrhage.

7. **Melanoma(Mel):** Melanoma is a malignant tumor that forms from melanocytes and comes in a variety of morphologies. If detected early enough, it can be treated with a simple surgical excision.

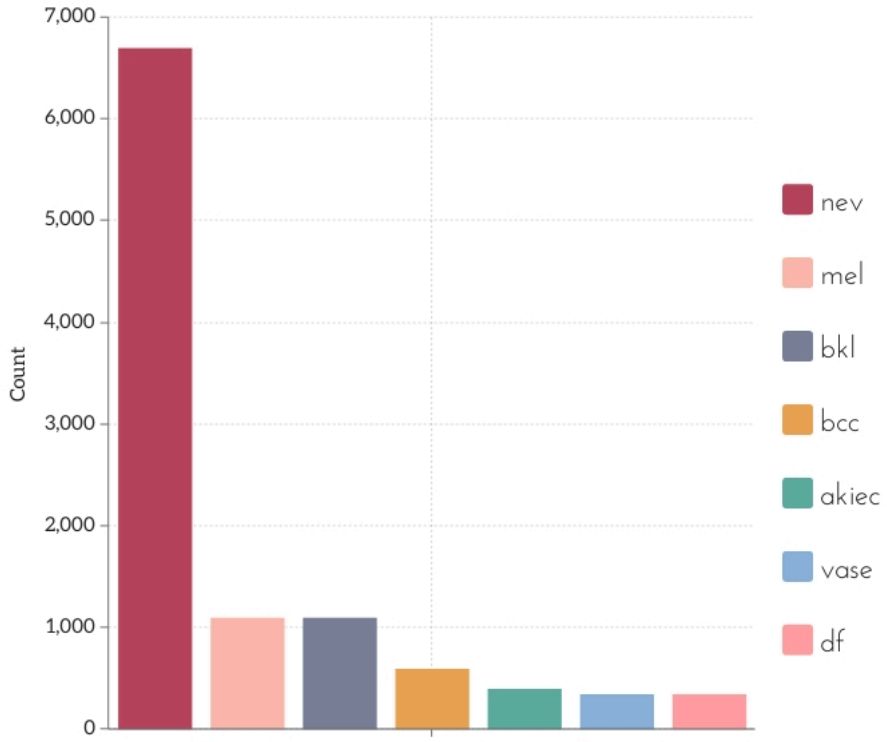


Figure 3.3: Data distribution table

3.2.2 Data Preprocessing:

HAM10000 contains raw data. Therefore, we transform it into an appropriate form to create an efficient model. This dataset includes 10,015 dermatoscopic images, and

some photos are blurry and far away from our interest. We removed those images. Data cleaning, feature and label selection, data transforms, and feature engineering was also performed. Everything was done by reading the CSV file containing the metadata. But the metadata didn't include the image files. So, to read the image files, we used glob functions and read the image path first. Using the image path, we read the image files one by one from the folders. The image size was (64,64,3). All the images were not in the same condition. To get a better result from our model all the images need to be in the same condition. If they are in different conditions the model will not perform well. Without proper preprocessing, the training time would be very high. Proper preprocessing makes the model more efficient. For preprocessing we have followed several steps. All the steps have been described below:

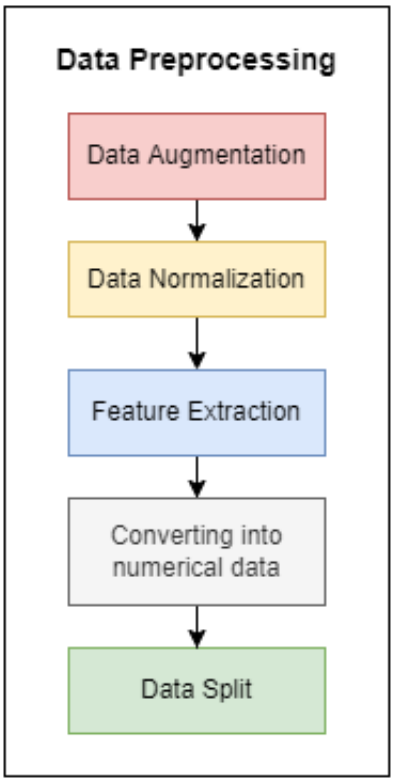


Figure 3.4: Steps of Data Preprocessing

Data Augmentation:

As we can see from the figure:1 'Data Distribution', most of the images belong to the NV class. From that, we can conclude that the dataset is very imbalanced. An imbalanced dataset will create a bias towards larger classes. To balance the dataset, we have augmented the dataset using the resampling technique. It generates a unique sampling distribution on the basis of the actual data. To resample it copies the same image again and again with different alignment or complexion. To do that it uses interpolation techniques. After augmentation, the frequency for all the classes was 1500.

Data Normalization:

Data normalization is a method of database planning that decreases data repetition, ensures data uprightness, and excludes undesirable features such as insertion, update, and deletion anomalies. There are normalizing techniques accessible, such as z-score normalization, min-max normalization, and decimal scaling normalization. We normalized the dataset by dividing the value of images pixels by 255. 255 is distinguished as the grayscale value of an image. After normalization, all the pixel values were between (0,1).

Feature extraction:

Finding out features is significant. Features are the inputs that produce the output or result. And the result of classification is known as a label. It makes the dataset more manageable for further analysis and processing. In this work, we also extracted the features, which helped to make the dataset more processable. Initially, the dataset has a large number of features. This large number of features can make the model slow. Using feature extraction we reduced the number of features. We selected the features based on their correlations with the ground truth value. We only kept the features which have a high correlation with the ground truth. It made our model much faster and efficient.

Converting into numerical data:

The ground truth column 'dx' is not encoded. For the sake of our model, we have encoded the dx column and the encoded labels were stored in a new column named 'label'. We have used the LabelEncoder function, which alters our labels into the numerical format. The python standard library provides the LabelEncoder function. If a dataset holds various scales for each feature then ML performs well.

Data Split:

We have split our dataset into two parts namely, training data and testing data. Training data is used to train the dataset onto. Whereas, testing data is used to test the performance of the model on the dataset. 75% of the data were training data and 25% were testing data. But in deep learning models we also need validation data. Validation data is used to tune the hyperparameters of the model. If we tune the hyperparameters on the training or test data it creates a bias towards the data. That is why to tune the hyperparameters we further split the training data into 25% validation data. Validation data is only used for tuning the hyperparameters. The training dataset is largest thus the model will be able to train on a large size of images which in turn will make the model stronger.

To reiterate, at first we trained our dataset on the training dataset. After being done with training we tuned the hyperparameters of each algorithm on the validation dataset. Lastly, to judge the performance of the models we applied the model into the testing dataset.

Chapter 4

Pretrained Transfer Learning Models

We used various pre-trained CNN models in our research using transfer learning. They are VGG16, VGG19, InceptionV3 and EffecientNet B7. The advantage of using transfer learning is that it is pre-trained on the Imagenet dataset which is one of the largest image datasets. Whereas, our HAM10000 dataset contains only 10000 images which is not enough to train a deep learning model. By using pre-trained models we can increase the effectiveness of a model. All of these models are different architectures of CNN and are trained on Imagenet. We will describe the models in detail in the paragraphs below:

4.1 VGG19

VGG19 is an all-inclusive CNN with pre-trained layers and a good grasp of visual shape, color and structure. VGG19 is a deep neural network trained on millions of images using various classification tasks. The VGG-19 consists of a 19-layer deep convolutional neural network. It is a combination of 16 convolutional layers, 5 max pool layers, 3 fully connected layers, and 1 softmax layer, as well as being a variation of the VGG model. The size and number of convolutional and fully connected layers are regulated by CNN's architect. For instance, A fixed size (224 * 224) RGB image was supplied as input to the VGG19 architecture, suggesting that the matrix was of shape (224 * 224 * 3).Only the average RGB value of each pixel in the entire training set was subtracted as a pre-processing step. Spatial padding was applied to keep the image's spatial resolution. Max pooling was done with stride 2 over a 2 * 2 pixel window. Since earlier models used tanh or sigmoid functions, the Rectified linear unit (ReLu) was used to express non-linearity to improve model classification and computational speed, and this proved to be considerably superior to them. Again, this model consisted of three fully connected layers, the first two of which were 4096 pixels in size, followed by a layer with 1000 channels for 1000-way ILSVRC (ImageNet Large-Scale Visual Recognition Challenge) classification, and lastly a softmax function. Moreover, The ability of VGG19's autonomous feature extraction to uncover the qualities that distinguish each cancer kind without having to spend time manually examining them makes it straightforward to do so. VGG19

is a great model with some drawbacks such as it was a rather large network in terms of the number of parameters that needed to be trained.

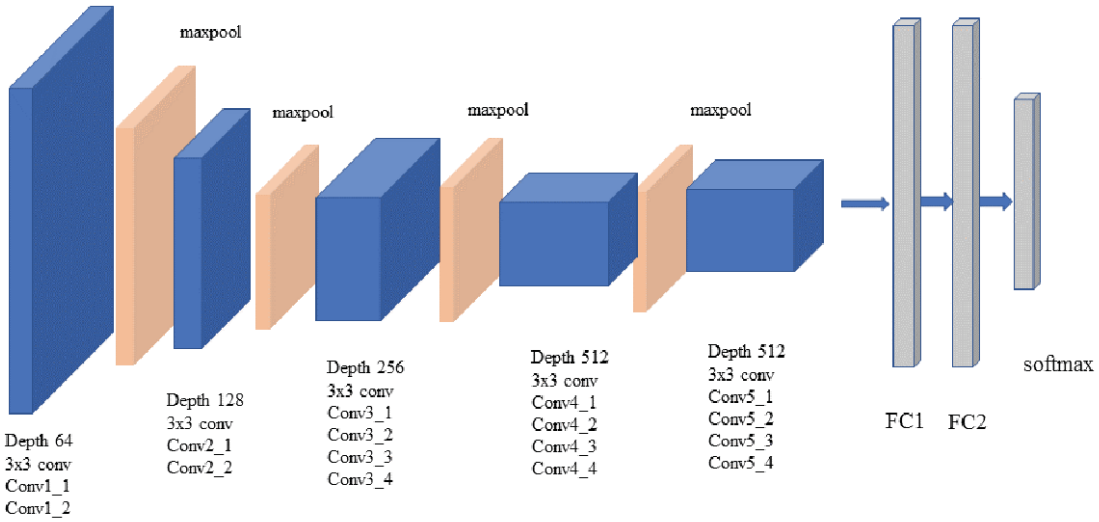


Figure 4.1: VGG-19 Architecture

4.2 VGG16

In 2013, Karen Simonyan and Andrew Zisserman proposed the VGG network and named it VGG after the Visual Geometry Group department at the University of Oxford to which they belonged, and in 2014 the actual model was submitted in the ImageNet Challenge. As it has 16 layers in its architecture, it is called VGG-16. Compared to other CNN models, it has achieved 92.7% accuracy while using the ImageNet dataset, and it contains 14 million images. First of all, this model recommends using small 3x3 filters(receptive field) with stride 1 all along with the network. Because it would provide the network the ability to merge more efficiently, for example, if we use five 3x3 layers, there will be five non-linear activation layers which will help the decision function to be more prejudiced towards its decision; as a result, the outcome will be more reliable. Also, throughout the architecture, it includes consistent padding, 2x2 Maxpool with stride 2. Secondly, it also reduces the amount of weight parameters in the model. After the convolutional layer stacks there are three fully connected layers in the architecture, where the first 2 layers have 4096 neurons each, and the last layer has 1000 neurons. Finally, the Softmax activation function is used in the output layer for categorical identification.

Many different and improved models have been introduced since VGG-16 was proposed, but VGG-16 keeps up the interest of researchers and data scientists because of its practical uses. Here are the use cases-

1. **Image Recognition or Classification:** It can be used in the medical sector to detect various diseases through X-ray or MRI imaging.

2. **Image Detection or Localization:** In terms of picture detection, it's quite efficient.

3. **Image Embedding Vectors:** The model may be used to train to build picture embedding vectors, which can be used for a situation where face verification is needed.

Along with the great features, this model has some drawbacks, such are-

1. **Edge Computation:** As this model has 138M parameters along with a size over 500MB, the time for edge computing is much higher, so it's quite impossible to presume the computing time.

2. **Vanishing or gradient problem:** To control the vanishing or gradient problem, there is no certain technique available.

To summarize, we can state that VGG-16 is an efficient convolutional neural network with a stacked convolutional layer of small filters(receptive fields) which bring a revolutionary impact on image detection or recognition.

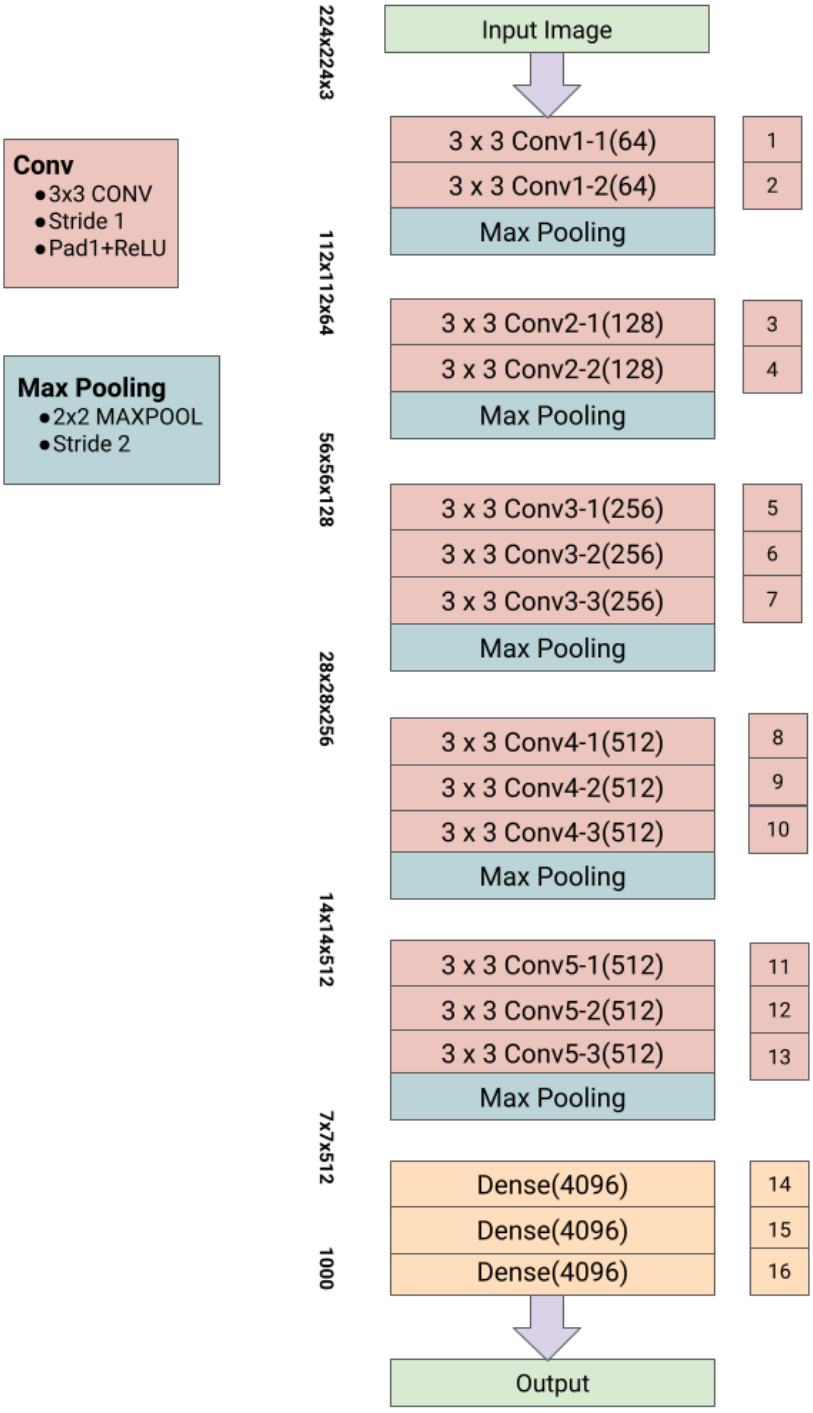


Figure 4.2: VGG-16 Architecture

4.3 Inception v3

The Inception network was an important landmark in the development of CNN classifiers. Before the inception model, most popular CNN models were stacked convolutional layers deeper and deeper for getting a more satisfactory performance. The Inception network, on the other hand, was complicated (laboriously engineered). It used a lot of schemes to push the performance both in terms of accuracy and speed. Its constant development showed the result of several versions of the network.

Inception v3 is the third edition of the Inception CNN model developed by Google. It is more accurate and has a lower computational cost than prior versions. It's a 48-layer pre-trained convolutional neural network model with a lower error rate than previous models. It is a version of the network already trained and it shows the accuracy of 78.1% from the ImageNet dataset. The ImageNet dataset is composed of more than one million pictures which are split into training datasets containing 1,281,167 pictures and evaluation datasets 50,000 pictures, respectively. This inception model can categorize pictures into 1000 object categories, such as a mouse, keyboard, pen, flowers, and many animals. Therefore the network has retained rich feature representations for a wide range of pictures. The significant modifications done on the Inception V3 model are Factorization into smaller convolutions, Spatial factorization into asymmetric convolutions, Utility of auxiliary classifiers, and Efficient grid size reduction. The model extracts general features from the input pictures in the first part and categorizes them based on those features in the second part. The input size of this model is 299*299*3 and the output size is 8*8*2048.

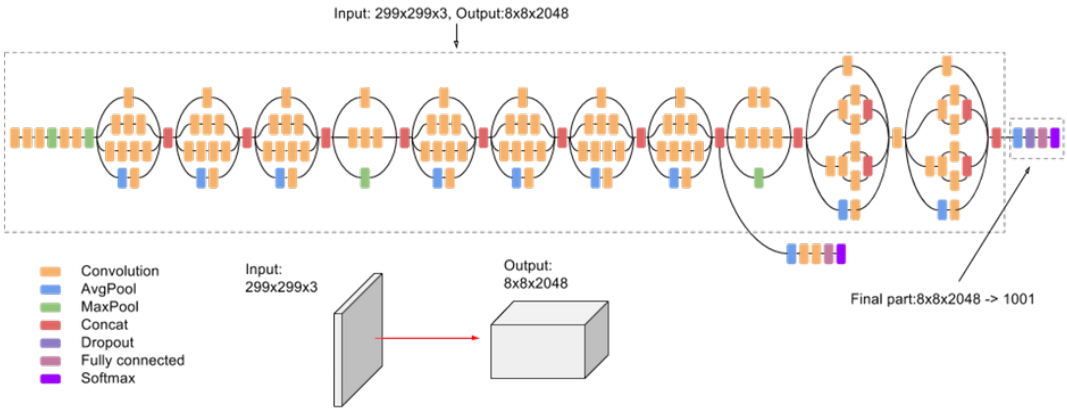


Figure 4.3: Inception v3 Architecture

4.4 EfficientNet B7

Mingxing Tan and Ruoming Pang from the google research team in 2019 first proposed the model EfficientNet B7. It is mainly a scaling method which uniformly scales all depths and resolution so that the model can lead to a better performance. They first noticed that scaling of the model's depth, resolution and other parameters can lead to a good result. This model used Compound scaling methods to scale the propagations of the network. The applied grid search strategy works under a

fixed resource obligation to find the relationship between the different scaling propagations of the baseline network. A baseline network was first designed by using the search technique. It enhances accuracy by floating point operations per second basis(FLOPS). We used the advanced version of EfficientNet which is cognizant as EfficientNet B7. It's special characteristic is the highest accuracy and still less number of parameters.

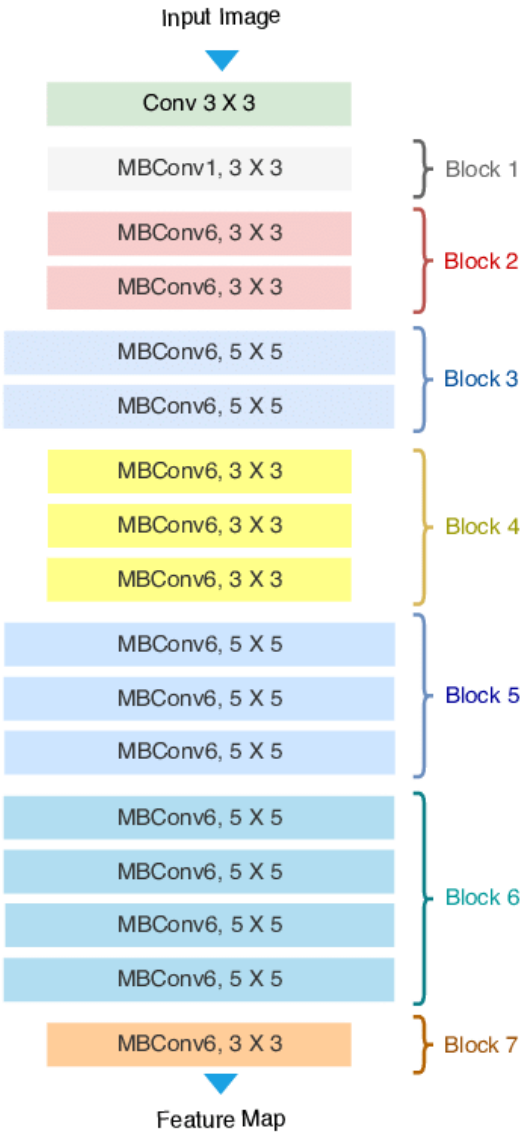


Figure 4.4: EfficientNet B7

4.5 ResNet 50

A residual neural network (ResNet) is a type of artificial neural network (ANN) that is based on components found in cerebral cortex pyramidal cells. Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun initially described it in their 2015 computer vision research article titled 'Deep Residual Learning for Image Recognition.'Residual neural networks employ skip connections to skip across some levels. Two or three layers of skips with nonlinearities (ReLU) and batch normalization are

typical ResNet models. Skip connections can be used in two ways. They begin by designing a new route for the gradient to follow in order to tackle the problem of vanishing gradients. They also make it possible for the model to learn identity function. This ensures that the model's top layers execute at the same level as its lower layers. Residual blocks, in essence, make learning identity functions for the layers more easier.As a result, ResNet boosts deep neural network performance by adding more neural layers and lowers the error rate. ResNet is available in a variety of configurations, each with a different number of layers but the same notion.ResNet-50 refers to a variant that can operate with up to 50 neural network layers. The ResNet-50 model contains a convolution and identity block for each level. Each convolution block has three convolution layers, and each identity block has three convolution layers. The ResNet-50 can train over 23 million parameters. Again, ResNet-50 is

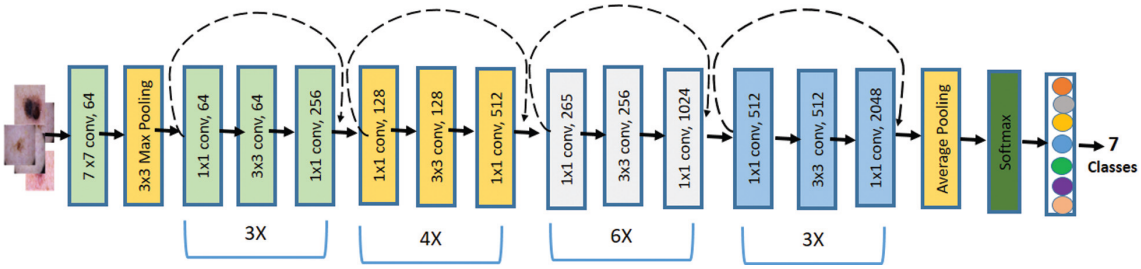


Figure 4.5: ResNet-50 Architecture

made up of 48 convolutional layers with 64 distinct kernels, [18] one max pool layer with a stride of size 2. To make a total of 9 layers, these layers are duplicated three times. The following layer has a variety of kernels and is replicated four times for a total of 12 layers. The next 49 levels are composed of different types of kernels that are repeated multiple times. As a result, the final layer of this structure is a properly networked layer with 1000 nodes and a softmax function, which gives us an average pool. This model may be loaded with a pre-trained version of the network that has been trained using photos from the ImageNet database.As a consequence, the network has a better understanding of feature representation for a wide range of pictures.

Chapter 5

Proposed Model

The purpose of the proposed skin cancer classification model is to classify skin cancer in an early stage with good accuracy which helps doctors to start treatment early. To do so, the model requires designing a process that takes data from images as an input, systematically processes input data, and produces predictions. Figure 2 provides a high-level view of the model design.

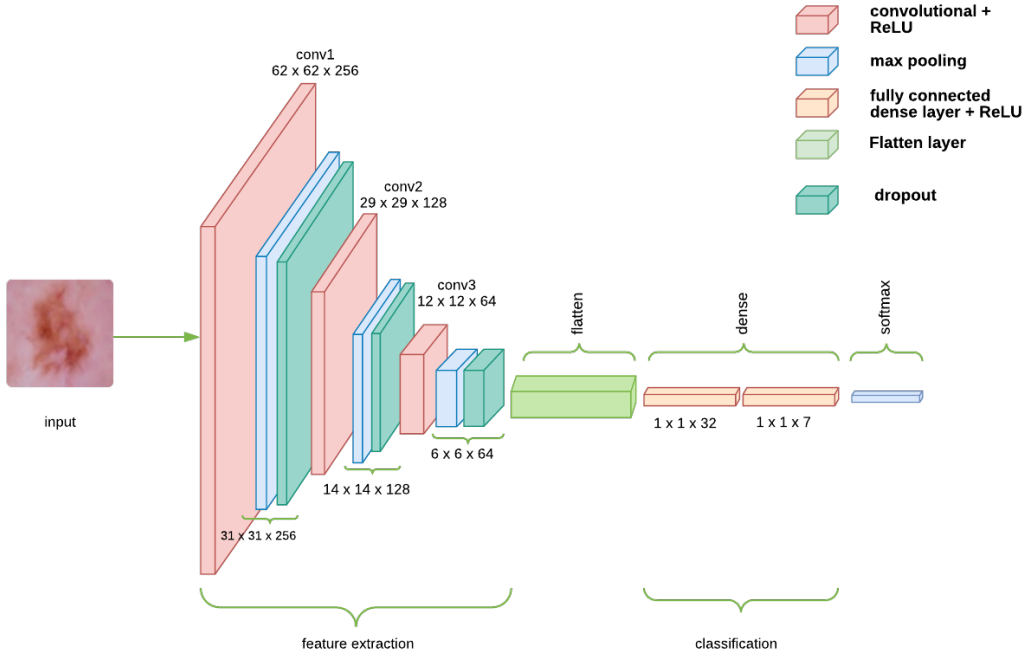


Figure 5.1: Proposed Model Architecture

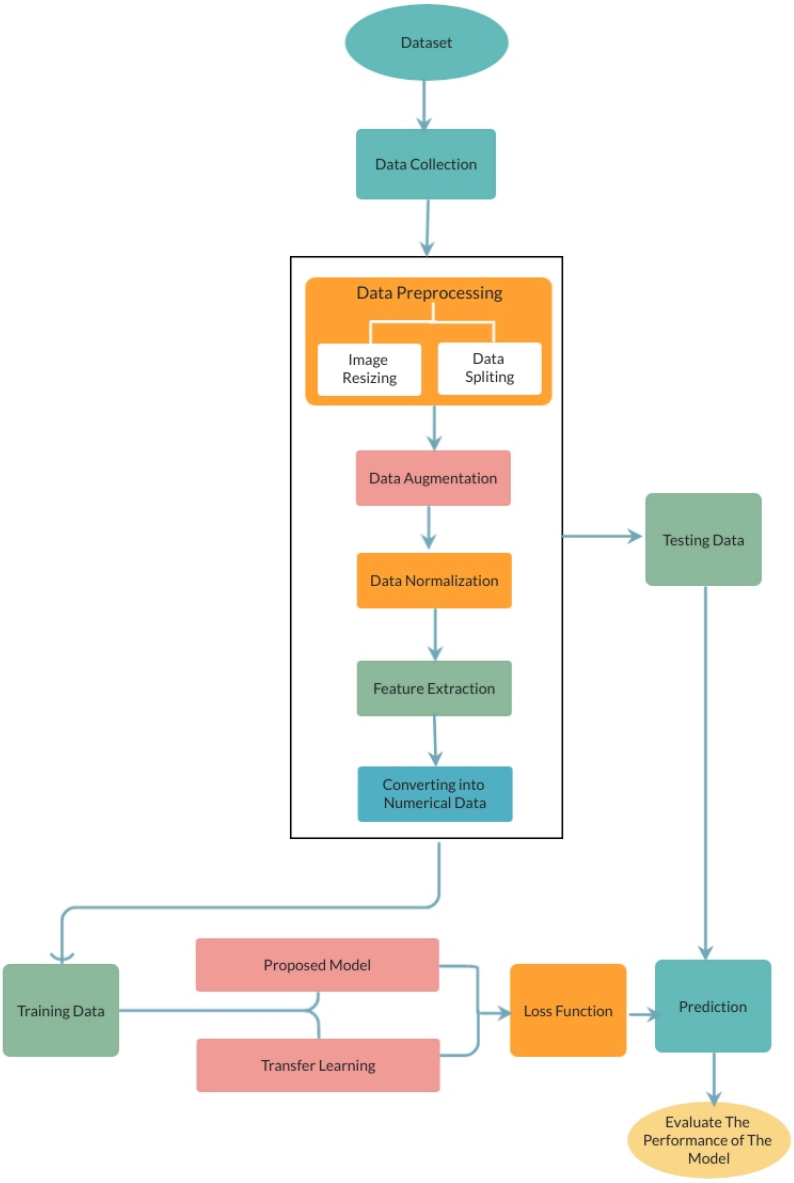


Figure 5.2: Workflow of our proposed model

Our proposed model in short has three convolutional layers. For each layer, we have used relu as the activation function and the input shape was (64*64). For each of the layers kernel size was (3,3). On the other hand, filter sizes for each layer are different. For the first layer filter size is 256, for the second layer it is 128 and for the last layer, it is 64. Between each layer, we added a pooling layer Pooling layer reduces the size while keeping the image intact. Pool size was determined at (2,2). At the end of all three layers, we have added a flatten layer to flatten the image. In the last dense layer, we used the Softmax activation function

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_3 (Conv2D)            (None, 62, 62, 256)       7168
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 31, 31, 256)       0
_________________________________________________________________
dropout_3 (Dropout)          (None, 31, 31, 256)       0
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 29, 29, 128)       295040
_________________________________________________________________
max_pooling2d_4 (MaxPooling2 (None, 14, 14, 128)       0
_________________________________________________________________
dropout_4 (Dropout)          (None, 14, 14, 128)       0
_________________________________________________________________
conv2d_5 (Conv2D)            (None, 12, 12, 64)        73792
_________________________________________________________________
max_pooling2d_5 (MaxPooling2 (None, 6, 6, 64)          0
_________________________________________________________________
dropout_5 (Dropout)          (None, 6, 6, 64)          0
_________________________________________________________________
flatten_1 (Flatten)          (None, 2304)              0
_________________________________________________________________
dense_2 (Dense)              (None, 32)                73760
_________________________________________________________________
dense_3 (Dense)              (None, 7)                 231
=================================================================
Total params: 449,991
Trainable params: 449,991
Non-trainable params: 0
```

Figure 5.3: Proposed model summary

5.1 Convolutional layer:

The convolutional layer is responsible for filtering the images using a number of kernels and passing the output to the next stage. It is invoked by the Conv2d function in python. Each kernel generates a unique image. The kernel slides over the input image and the dot product is taken from it. The main objective of this convolutional layer is to extract the feature map of the image. We get the feature map from the above species feature map. Later, this feature map is fed to the other layer[12].

5.2 Max Pooling layer:

Ordinarily, a convolutional layer is inherited by a pooling layer. Feature maps extracted from a convolutional layer can be very large in size, which increases the computational cost. So, it makes the operation slower. The pooling layer is used to decrease the size of the feature map so that the computational cost decreases and the operation becomes faster. There are several types of pooling methods depending on the model.

5.3 Dropout layer:

Dropout is used as a training strategy. By using this, some randomly chosen neurons are rejected. This means that any weight changes are not conveyed to the neuron on the reverse trip, and their influence to downstream neuron activity is erased temporally on the forward pass. Dropout is solely utilized during model training and is not used to assess the model's ability. A Simple Process for Preventing Overfitting in Neural Networks, 2014. Since the outputs of a layer under dropout are arbitrarily subsampled, the capability of the network is narrowed or lightened during training. As a result, when utilizing dropout, a larger network, e.g. more nodes, may be needed.Moreover, Dropout is a strategy for reducing the network's overfitting to the training data. Dropout can be used after convolutional layers and pooling layers. Dropout is frequently employed after the pooling layers, however this is only a guideline. When the dropout rate is less than a certain threshold, the accuracy gradually improves as the loss decreases. When dropout levels rise over a particular point, the model loses its ability to fit correctly.

5.4 Flatten layer:

Flatten layer and dense layer is the last layer of CNN. The output of the convolutional layer is in multi-dimensional shape but the dense layer requires a single-dimensional shape as input. So, in between convolutional layers and dense layers we need to use the Flatten layer to convert the multidimensional matrix to a single dimensional matrix as cubic or rectangular shapes can't be direct inputs to the next layer.

5.5 Dense layer:

It is a neural network layer which is densely connected, meaning all the neurons of each layer are connected with its previous layer's neuron. The image will be passed through all convolutional layers and pooling layers before being forwarded to the dense layer. Through a flatten layer we will convert the multidimensional output into one-dimensional output and pass it to the dense layer as input. It is a layer of neurons where all the neurons will receive input from the previous layer's neurons. Based on the output of the convolutional layer, the dense layer classifies the image. In the Neural Network each layer comprises neurons that calculate the weighted average of the input and pass it to a nonlinear function which is known as an "activation function" and the output of this function is considered as the neuron's output. The last layer of the dense layer is considered as output of the image. Moreover, the output layer has a softmax activation function. When there are 2 or more classes we need this function. Each neuron is associated with a certain class. For all classes each neuron will return probabilities of the input image respectively. Highest probability of a class will be considered as output.

5.6 ReLU Activation Function:

The activation function determines which neuron will be activated first. The most used activation function is Relu. RELU (Rectified Linear Unit) is a non-linear activation function that is becoming extremely prevalent. Multi-layer neural networks and deep neural networks are the most widely used applications. The highest value between zero and the input value is the output of ReLU, according to the equation of relu. When the input is negative, the output is zero, and when the input is positive, the output is equal to the input. The vanishing gradient problem is solved by ReLU. When the number of layers increases, the ReLU function does not induce the vanishing gradient issue. It brings a sharp or drastic change when there is a different feature. All the negative values are converted into 0 so there are no negative values available. Lastly, since the derivative of the ReLU function is 1 for positive input, it can speed up the training of deep neural networks especially in comparison to typical activation functions.

The mathematical equation of ReLU is:

$$RELU(x) = \begin{cases} 0 & if\ x < 0 \\ x & if\ x \geq 0 \end{cases}$$

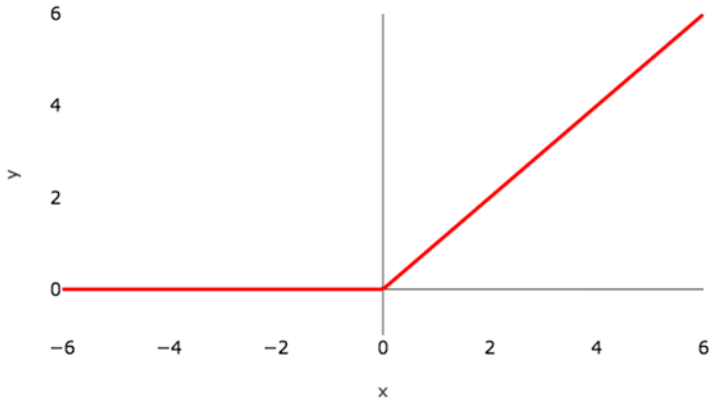


Figure 5.4: graph of an activation function ReLU

5.7 Loss function:

Loss functions are used to improve the algorithms of machine learning and It calculates the gradients. In simple words, it predicts how reliable the model is in prediction. To calculate the loss there are some functions which are called Loss function. It calculates the difference between the predicted output and the algorithm's current output. The loss will be the lowest if the model predictions are the closest to the actual data but the loss will be the highest if the predictions are completely different from the actual data. The loss is computed based on the

training and validation data. It is categorized in 2 sets and those are Classification (Discrete Values) and Regression (Continuous Values). It is dependent on how well it performs in these two sets.
We have used one loss function in our model which is Categorical Cross-Entropy. To compute the loss by Categorical Cross-Entropy, following formula can be used-

$$Loss = -\sum_{i=1}^{Output\ size} = yi.log\ \hat{y}i$$

Here,
$\hat{y}i = i$-th scalar value in the model output
$yi =$ The corresponding target value,
output size = The number of scalar values in the model.
The negative sign assures that the loss gets lower.

5.8 Adamax optimizer:

The Adamax algorithm is a version of the Adam algorithm that uses the infinity norm. In some cases, Adamax outperforms Adam, particularly in models containing embeddings.Individual weights in Adam are updated by scaling their gradients inversely proportional to a (scaled) L2 norm of their current and prior gradients. The L2 norm-based update rule may be generalized to a Lp norm-based update rule. For larger p, such versions become numerically unstable[5].The ut value in Adam with 1 has been shown to merge to a gradually stable value.

$$u_t = \beta_2^{\infty}.u_{t-1} + (\beta_2^{\infty}).\,|g_t|^{\infty} = max(\beta_2.u_{t-1}, |g_t|)$$

Chapter 6

Performance Evaluation Matrices of CNN Models

6.1 Performance Metrics:

We have used several CNN architectures for our model. Not all of them perform on the same scale. Some performed better than other models. The performance of these architectures are estimated through Accuracy. In this chapter, we first described the performance metrics and later we disclosed the performance of each CNN architecture's performance based on the abovementioned metrics.

6.1.1 Accuracy:

Accuracy is a performance metric used to assess a classification machine learning model. The accuracy of a machine learning model is a metric for determining which model is the best at predicting a label by analyzing the parameters.

$$Accuracy = \frac{True_{positive} + True_{negative}}{True_{positive} + True_{negative} + False_{positive} + False_{negative}}$$

6.1.2 Precision:

Precision indicates the number of true positives divided by the total number of positive predictions. It measures how well a model classifies a sample as positive. When the model generates a large number of erroneous Positive classifications or a small number of accurate Positive classifications, the denominator increases and the accuracy decreases. The precision is high when the model makes a significant percentage of correct Positive classifications and fewer incorrect Positive classifications.

$$Precision = \frac{True_{positive}}{True_{positive} + False_{positive}}$$

6.1.3 Recall:

Recall determines the model's ability to classify positive data points. The higher the recall, the greater number of positive samples are being classified. Negative classifications do not affect recall.

$$Recall = \frac{True_{positive}}{True_{positive} + False_{negative}}$$

Chapter 7

Experimental Result and Performance Analysis

7.1 Experimentation

We tried a twofold approach in this experiment, which quickly helped us reach our goal. We compared the transfer learning models with our proposed model and found a better model for classifying skin cancer in an early stage of cancer. Since cancer is a deadly disease, finding out the cancer type in an early stage might be very helpful for getting excellent treatment. Furthermore, comparing the model using various performance metrics also helped us evaluate the model more accurately on different scales. After completing the data preprocessing, we have implemented the transfer learning models such as VGG16, VGG19, inception v3, ResNet 50, and EfficientNet B7. We used the same batch size and epochs for the experiment to better comprehend the performance. Our research has built a CNN model for classifying skin cancer more precisely. Firstly, we find a good combination of convolution and max-pooling layers to find out the features. We have then figured out the right amount of dense layer for the classification. Later we also add a dropout for achieving high accuracy. We used an Adamax optimizer in this model. Lastly, we figure out an excellent optimizer for our problem. We observed the performance of different popular optimizers such as Adam, Adadelta, SGD, RMSProp, Adagrad, and Adamax. Among these six other optimizers, we noticed that adamax optimizers performed well. It gave us the highest accuracy with batch size 16 epoch size 50 and drop off 0.3, which is shown in the table. We did not use any batch normalization. Though, batch normalization helps a model learn more independently and efficiently. Moreover, this layer also helps a model to avoid overfitting. However, our model avoided overfitting without adding batch normalization as the training and validation curves are converging. Furthermore, as a research purpose, we experimented. Figure and figure describe the training and validation accuracy and loss curve where we can see that before adding the batch normalization, the model is more stable, which means less fluctuation on the curve. After adding the batch normalization, the validation curve fluctuation happens because of the availability of sufficient training data. However, the accuracy is higher after adding the batch normalization layer. That is why we did not add batch normalization. So, we have chosen the Adamax optimizer for our skin cancer classification problem. This optimizer helps to train our model very quickly.

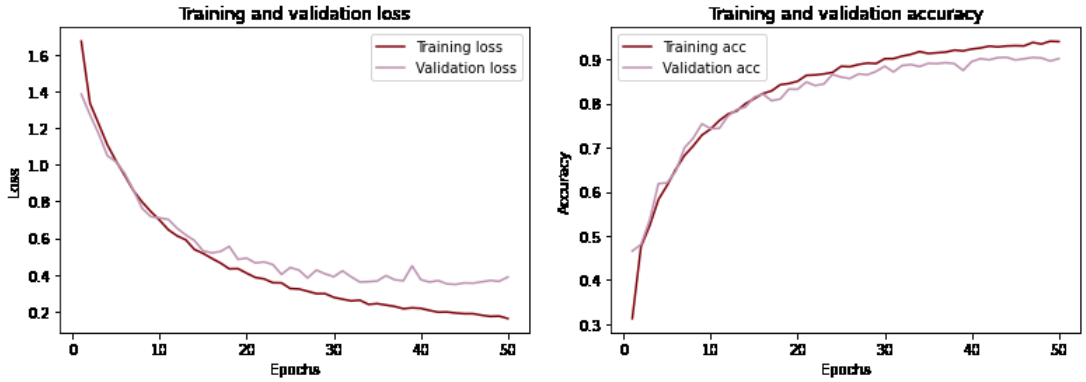


Figure 7.1: Before adding batch normalization

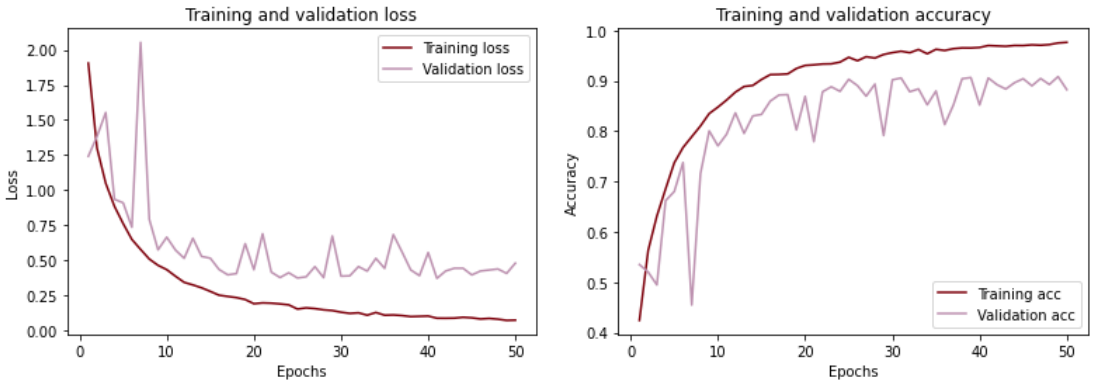


Figure 7.2: After adding batch normalization

optimizer	validation accuracy	Testing accuracy	batch size	epochs	dropoff
Adam	0.93448	0.85987	16	50	0.3
Adadelta	0.27780	0.25752	16	50	0.3
SGD	0.87252	0.80571	16	50	0.3
RMSProp	0.86541	0.78590	16	50	0.3
Adagrad	0.56932	0.54323	16	50	0.3
Adamax	0.95987	0.87999	16	50	0.3

Table 7.1: Different optimizers performance analysis

We have also run experiments on batch size, for our dataset we observed that, among 8,16,32 batch size 8 gave us the highest accuracy with 50 epochs and 0.3 drop off. Table 7.2 demonstrates the result of the experiment result.

optimizer	validation accuracy	Testing accuracy	batch size	epochs	dropoff
Adamax	0.97613	90.05	8	50	0.3
Adamax	0.95987	0.87999	16	50	0.3
Adamax	0.9537	0.8780	32	50	0.3

Table 7.2: Performance of proposed model with different batch size

In this similar fashion we tuned some other parameters and tried to figure out the best-fitted parameters for our proposed model which is demonstrated in the following table 7.3

optimizer	validation accuracy	Testing accuracy	batch size	epochs	dropoff
Adamax	0.9867	0.9055	8	75	0.3

Table 7.3: proposed model's performance summary table

After these experiments we came up with a CNN model which gives us the best result in the HAM10000 dataset. The summary of our cnn model shown in the figure.

7.2 Performance analysis of different models

In this section, we compare and contrast the performance of our proposed model and the pre-trained transfer learning model such as VGG16, VGG19, EfficientNet B7, Inception v3, ResNet 50. From the experimental results shown in table and figue we can deduce that the performance of our proposed model is better than the pre-trained transfer learning models. Figure defines a bar graph on the accuracy of the CNN models.

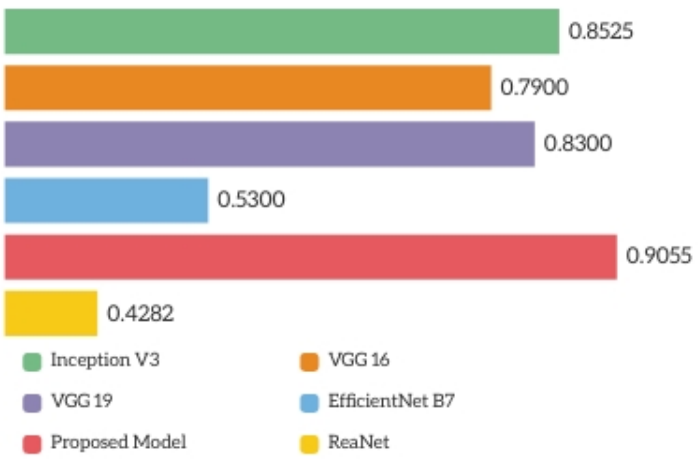


Figure 7.3: Bar graph on the accuracy of different CNN models

Our proposed model has accomplished the highest accuracy of 90.55%. The accuracy achieved by other transfer learning pre-trained models are such that: Inception v3 was 85.25%, VGG19 was 83%, VGG16 was 79% , Efficient Net was 53% and ResNet was 42.85%. The performance of efficient net was not upto the marks because the amount of data was that sufficient the model. The efficient net model is so deep that it requires a large data set to perform well. However the transfer learning model has faster convergence than our proposed model. Among all the models our proposed model has the lowest number of layers: only 12, vgg16 has 16 layers and vgg19 has 19 layers, inception v3 has 48 layers and efficient net has 813 layers and ResNet is 50 layers deep. For our problem we believe that 12 layers are good enough. Unnecessary layers could be deadly and will increase the number of parameters for a small dataset. However, it should be noted that smaller and medium layers CNN models perform better in typical dataset and Larger layers CNN models perform better in huge datasets.

Models	**Accuracy**	**Precision**	**Recall**
Inception V3	85.25%	84.9%	80%
VGG16	79%	65.67%	68.79%
VGG19	83%	68.54%	69.45%
EfficientNet B7	53%	N/A	N/A
ResNet50	42.85%	78.6%	77%
Proposed Model	90.55%	92.89%	91.85%

Table 7.4: Comparison between proposed model and other pre trained model.Training and Validation Accuracy and Loss, In our study we only find out the accuracy,precision and recall of our proposed model. For other transfer learning models we find out only accuracy. In paper [20] the author finds out precision and recall of inception v3 and resnet50 on the same dataset. And VGG16 and VGG19 precision and recall attained from[14]

7.3 Training and Validation Accuracy and Loss

Inception V3

Using the Inception V3 model we attained a testing accuracy of 85.25% and validation accuracy of 97.25%. Figure 7.4 and Figure 7.5 represents training and validation accuracy and loss of inception v3 model. In the training and validation loss graph we perceived that the loss decreased enough over time which means that there is no underfitting. We also observed that the training and validation curves are converging, hence we can say that there is no overfitting in Inceptionv3 on the HAM10000 dataset. The fluctuation is also less which defines that the amount of data is sufficient for this model.

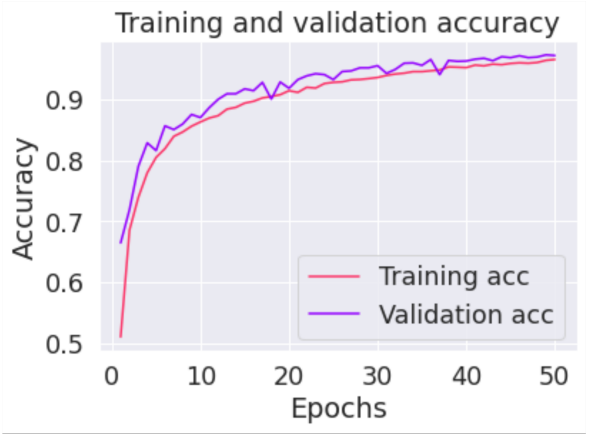


Figure 7.4: Training and validation accuracy for inception v3

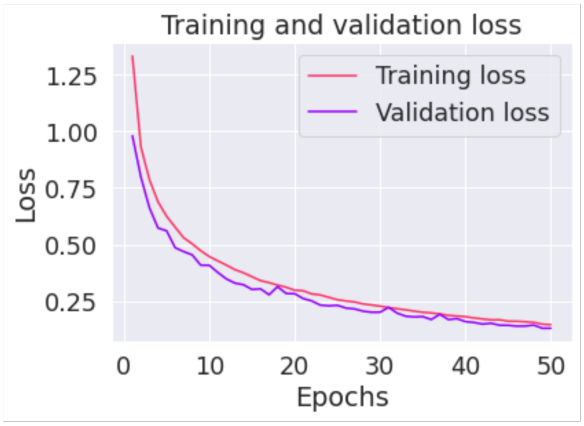


Figure 7.5: Training and validation loss for inception v3

VGG19

The VGG19 model attained a testing accuracy of 83% and validation accuracy of 89%. Figure 7.6 and Figure 7.7 represents training and validation accuracy and loss of VGG19 model. In the training and validation loss graph we perceived that the loss decreased enough over time which means that there is no underfitting. We also observed that the training and validation curves are converging, hence we can say that there is no overfitting in VGG19 on the HAM10000 dataset. The fluctuation is also less which defines that the amount of data is sufficient for this model.

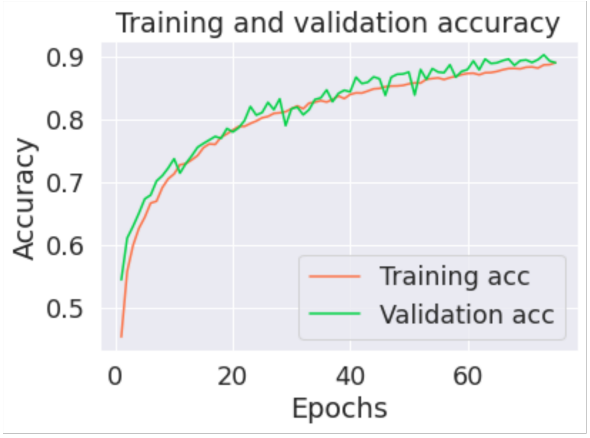


Figure 7.6: Training and validation accuracy for VGG19

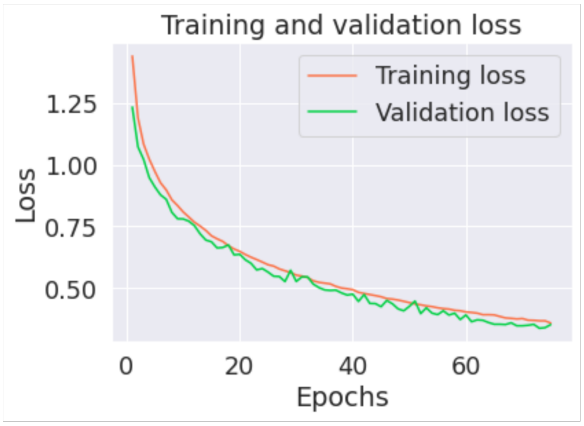


Figure 7.7: Training and validation loss for VGG19

VGG16

The VGG16 model attained a testing accuracy of 79% and validation accuracy of 86%. Figure 7.8 and Figure 7.9 represents training and validation accuracy and loss of VGG16 model. In the training and validation loss graph we perceived that the loss decreased enough over time which means that there is no underfitting. We also observed that the training and validation curves are converging, hence we can say that there is no overfitting in VGG16 on the HAM10000 dataset. The fluctuation is also less which defines that the amount of data is sufficient for this model.

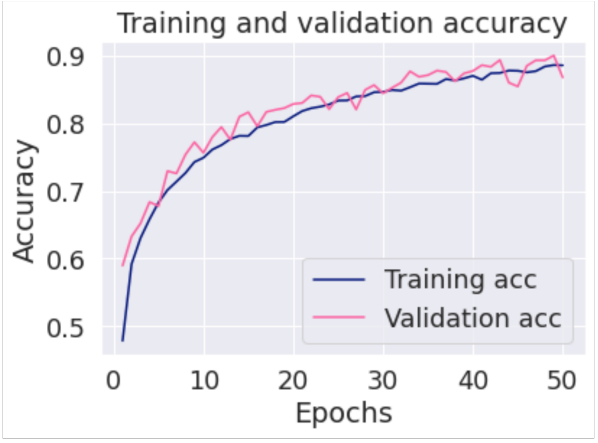


Figure 7.8: Training and validation accuracy for VGG16

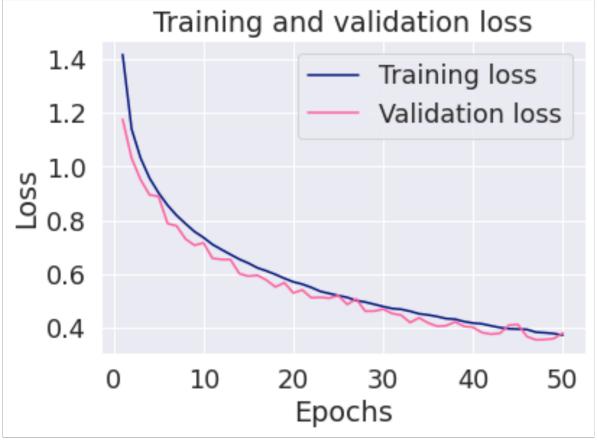


Figure 7.9: Training and validation loss for VGG16

EfficientNet B7

The EfficientNet B7 model attained a testing accuracy of 53% and validation accuracy of 54%. Figure 7.10 and Figure 7.11 represents training and validation accuracy and loss of efficient net b7 model. In the training and validation loss graph we perceived that the curve fluctuated too much. It means that the amount of data is not sufficient for that model of underfitting. We also observed that the training and validation curves are not converging till 50 epochs. This model required more epochs to reach its convergence point.

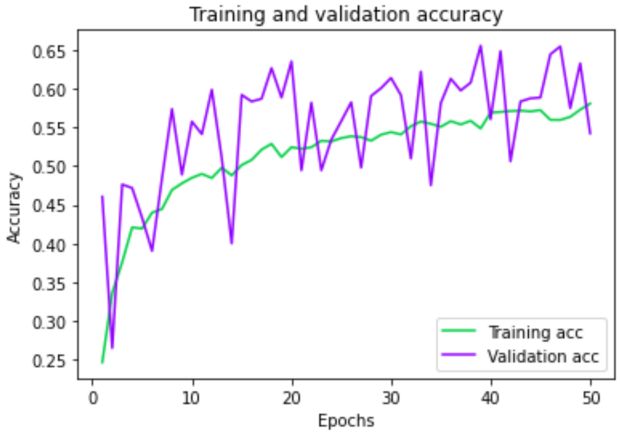


Figure 7.10: Training and validation accuracy for EfficientNet B7

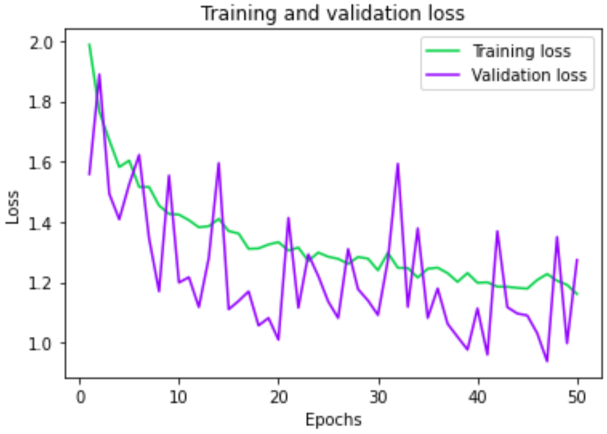


Figure 7.11: Training and validation loss for EfficientNet B7

ResNet 50

The ResNet 50 model attained a testing accuracy of 42.82% and validation accuracy of 44.96%. Figure 7.12 and Figure 7.13 represents training and validation accuracy and loss of ResNet 50 model. In the training and validation loss graph we perceived that the curve fluctuated too much. It means that the amount of data is not sufficient for that model of underfitting. We also observed that the training and validation curves are not converging till 50 epochs. This model required more epochs to reach its convergence point. However, in the training and validation loss curve the loss was decreasing over time which defines that there is no underfitting for this model on this dataset.

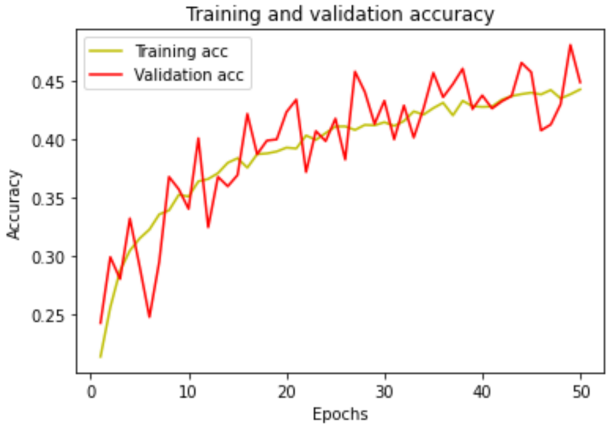


Figure 7.12: Training and validation accuracy for ResNet 50

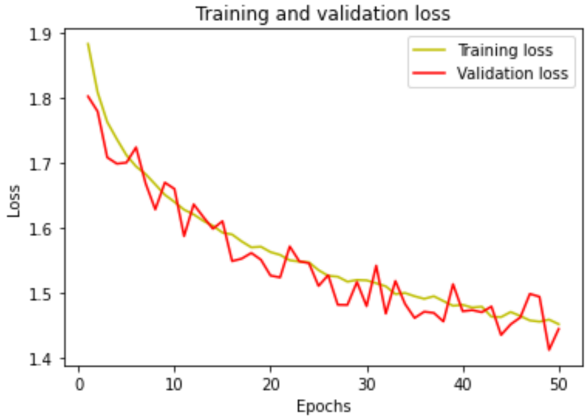


Figure 7.13: Training and validation loss for ResNet 50

Proposed CNN model

Our proposed model attained a testing accuracy of 98.67%and validation accuracy of 90.55%. Figure 7.14 and Figure 7.15 represents training and validation accuracy and loss of the model. In the training and validation loss graph we perceived that the loss decreased enough over time which means that there is no underfitting. We also observed that the training and validation curves are converging, hence we can say that there is no overfitting in the model on the HAM10000 dataset. The fluctuation is also very less which defines that the amount of data is sufficient for this model.

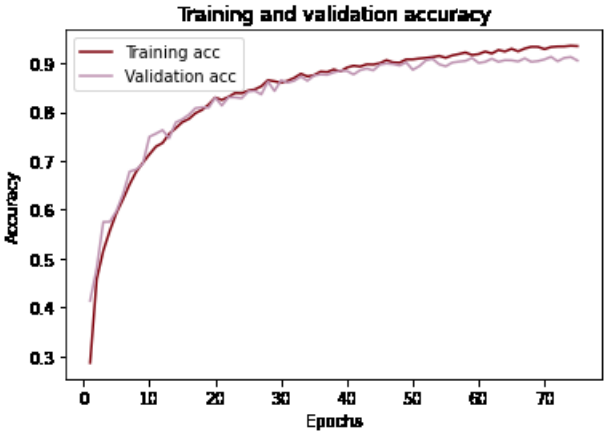


Figure 7.14: Training and validation accuracy of our Proposed CNN model

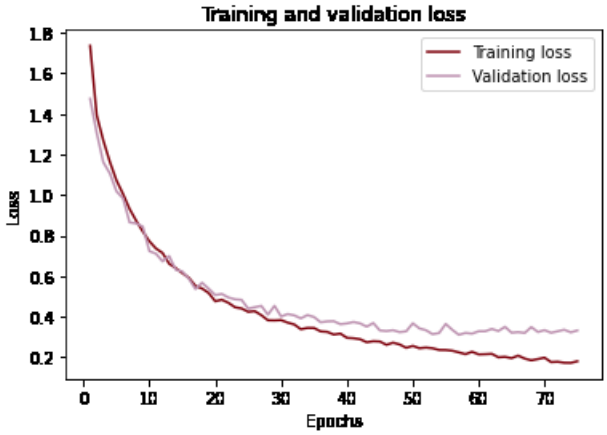


Figure 7.15: Training and validation loss of our Proposed CNN model

fraction classified incorrect graph

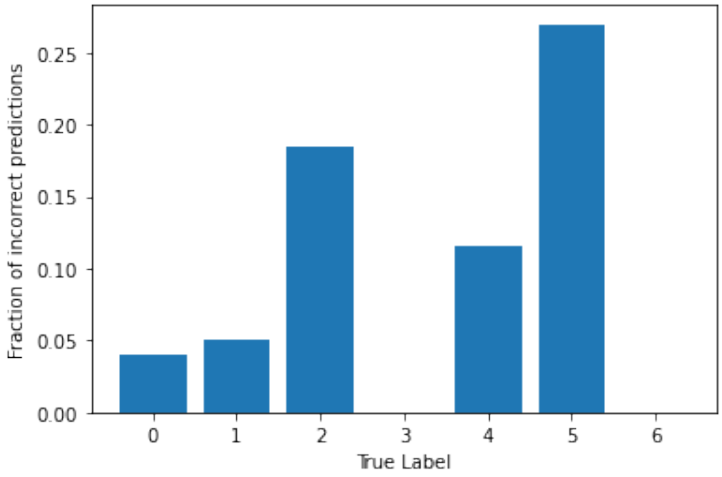


Figure 7.16: fraction classified incorrect graph

Chapter 8

Accuracy Comparison on Related Works

In this section, a comparison on the accuracy of our proposed model and related recent papers on the same dataset(HAM10000) as mentioned in the literature review section are established. The dataset used in this research has training set's 10015 dermatoscopic images gathered over a 20-year period from two separate locations: the Department of Dermatology at the Medical University of Vienna, Austria, and Cliff Rosendahl's skin cancer practice in Queensland, Australia. Dermatoscopy is a frequently used diagnostic procedure that, when compared to unaided eye inspection, enhances the identification of benign and malignant pigmented skin lesions[15]. In 2019 in a research paper [18], the author proposed a CNN model for identifying skin cancer using the same dataset as ours. Their maximum accuracy was 80%. Again in the same year (2019) another talented author classified skin cancer using a pyTorch pretrained model known as DenseNet121 model in their paper[5]. Their maximum accuracy was 85% in the 1st step and 75% in the second step. Table 8.1 demonstrates a summary of the accuracy review achieved by this paper and other research works recently done on this particular dataset that is being used in this paper. It is observed that our proposed CNN model attained the highest accuracy of 90.55% compared to the other methods. Figure x illustrates a bar graph on this accuracy.

Approaches	**Models**	**Accuracy**
This paper	CNN	90.55%
Dorin Moldovna[8]	CNN(DenseNet121)	85% and 75%
A A Nugroho, I Slamet, Sugiyanto [9]	CNN	80%

Table 8.1: Accuracy comparison between related works on the same dataset

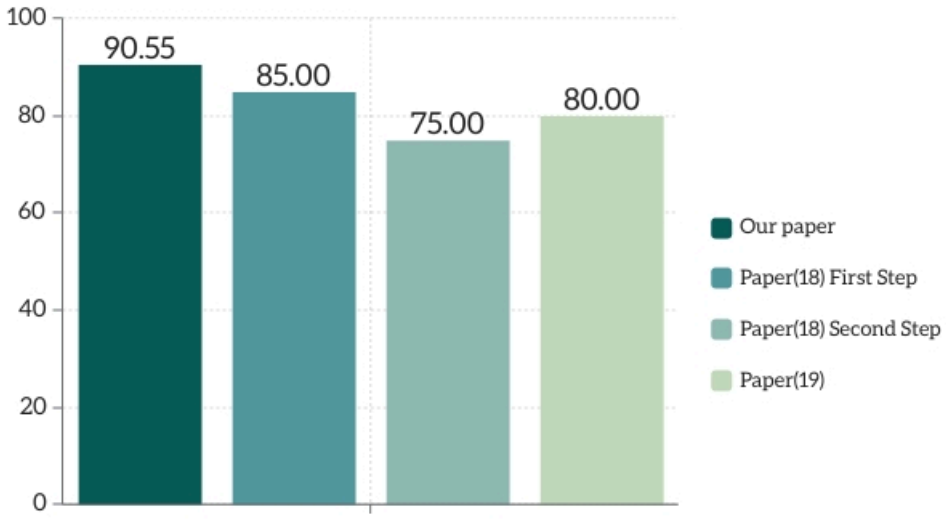


Figure 8.1: Bar Graph on Accuracy review

Chapter 9

Conclusion

The main motivation behind our work was to make the doctor's and patient's life easier by classifying the skin cancer type in earlier stages because the incidence rates of skin cancer have been rising enormously . Early classification can reduce the treatment cost and make the doctor's work efficient. Every year a lot of people die from lack of equitable recognition. Despite the fact that several studies have been conducted in the past to classify skin cancer, none of them have been successful in extending their research to multiple classes of skin cancer with high performance. We believe our easy method will help to reduce the rate of fatality. In our work, we used five predefined Transfer learning methods which are VGG-16, VGG-19, Inception v3, EfficientNet B7, ResNet 50 and our proposed method classifiers to classify skin cancer using the HAM10000 dataset. Later we compared the performance of the other models and our proposed model. After comparing models, We observed that our proposed method achieved 90.55% accuracy, the precision of our proposed model is 92.89% and recall is 91.85% which is the highest accuracy among all the models. Our proposed model performs better than the predefined transfer learning methods.

9.1 Future work

In our future work, we will improve our proposed model's accuracy, precision and recall and will try to reduce our computation time. As we already know that our dataset consists of seven types of skin cancer. That's why we plan to extend the scope of our model to classify more types of skin cancer. Further, we'll apply a color-constancy method to filter out the useless information and we will apply an image pre-processing technique which will remove hair, shades and glare from the image and it will help to recognize characteristics such as shape, color, size, and texture more precisely. Later, we have plans to integrate our proposed model with Inception V3 to make the performance better. Furthermore, We are planning to make an mobile application through which a user can upload skin lesion images and get a primary idea about the type of the lesion.

Bibliography

[1] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[2] T. DeVries and D. Ramachandram, "Skin lesion classification using deep multi-scale convolutional neural networks," *arXiv preprint arXiv:1703.01402*, 2017.

[3] A. H. Shahin, A. Kamal, and M. A. Elattar, "Deep ensemble learning for skin lesion classification from dermoscopic images," in *2018 9th Cairo International Biomedical Engineering Conference (CIBEC)*, IEEE, 2018, pp. 150–153.

[4] P. Tschandl, C. Rosendahl, and H. Kittler, "The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions," *Scientific data*, vol. 5, no. 1, pp. 1–9, 2018.

[5] P. Dwivedi, *Understanding and coding a resnet in keras*, 2019.

[6] Y. Luan and S. Lin, "Research on text classification based on cnn and lstm," in *2019 IEEE international conference on artificial intelligence and computer applications (ICAICA)*, IEEE, 2019, pp. 352–355.

[7] A. Mahbod, G. Schaefer, C. Wang, R. Ecker, and I. Ellinge, "Skin lesion classification using hybrid deep neural networks," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2019, pp. 1229–1233.

[8] D. Moldovan, "Transfer learning based method for two-step skin cancer images classification," in *2019 E-Health and Bioengineering Conference (EHB)*, IEEE, 2019, pp. 1–4.

[9] A. A. Nugroho, I. Slamet, and Sugiyanto, "Skins cancer identification system of haml0000 skin cancer dataset using convolutional neural network," in *AIP Conference Proceedings*, AIP Publishing LLC, vol. 2202, 2019, p. 020 039.

[10] N. Aburaed, A. Panthakkan, M. Al-Saad, S. A. Amin, and W. Mansoor, "Deep convolutional neural network (dcnn) for skin cancer classification," in *2020 27th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, IEEE, 2020, pp. 1–4.

[11] Y. N. Fu'adah, N. C. Pratiwi, M. A. Pramudito, and N. Ibrahim, "Convolutional neural network (cnn) for automatic skin cancer classification system," in *IOP Conference Series: Materials Science and Engineering*, IOP Publishing, vol. 982, 2020, p. 012 005.

[12] M. Gurucharan, "Basic cnn architecture: Explaining 5 layers of convolutional neural network," *URL: https://www. upgrad. com/blog/basic-cnn-architecture*, 2020.

[13] H. K. Kondaveeti and P. Edupuganti, "Skin cancer classification using transfer learning," in *2020 IEEE International Conference on Advent Trends in Multidisciplinary Research and Innovation (ICATMRI)*, IEEE, 2020, pp. 1–4.

[14] N. Rezaoana, M. S. Hossain, and K. Andersson, "Detection and classification of skin cancer by using a parallel cnn model," in *2020 IEEE International Women in Engineering (WIE) Conference on Electrical and Computer Engineering (WIECON-ECE)*, IEEE, 2020, pp. 380–386.

[15] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020.

[16] S. Abd ElGhany, M. R. Ibraheem, M. Alruwaili, and M. Elmogy, "Diagnosis of various skin cancer lesions based on fine-tuned resnet50 deep network," *CMC-COMPUTERS MATERIALS & CONTINUA*, vol. 68, no. 1, pp. 117–135, 2021.

[17] M. S. Ali, M. S. Miah, J. Haque, M. M. Rahman, and M. K. Islam, "An enhanced technique of skin cancer classification using deep convolutional neural network with transfer learning models," *Machine Learning with Applications*, vol. 5, p. 100 036, 2021.

[18] Ü. Atila, M. Uçar, K. Akyol, and E. Uçar, "Plant leaf disease classification using efficientnet deep learning model," *Ecological Informatics*, vol. 61, p. 101 182, 2021.

[19] M. Dildar, S. Akram, M. Irfan, *et al.*, "Skin cancer detection: A review using deep learning techniques," *International journal of environmental research and public health*, vol. 18, no. 10, p. 5479, 2021.

[20] R. Garg, S. Maheshwari, and A. Shukla, "Decision support system for detection and classification of skin cancer using cnn," in *Innovations in Computational Intelligence and Computer Vision*, Springer, 2021, pp. 578–586.

[21] Y. Jusman, I. M. Firdiantika, D. A. Dharmawan, and K. Purwanto, "Performance of multi layer perceptron and deep neural networks in skin cancer classification," in *2021 IEEE 3rd Global Conference on Life Sciences and Technologies (LifeTech)*, IEEE, 2021, pp. 534–538.

[22] S. S. Nisha and M. N. Meeral, "Applications of deep learning in biomedical engineering," in *Handbook of Deep Learning in Biomedical Engineering*, Elsevier, 2021, pp. 245–270.