VISVESVARAYA TECHNOLOGICAL UNIVERSITY
Jnana Sangama, Belagavi-590018, Karnataka

A PHASE-2 PROJECT REPORT
ON

"COMPUTER AIDED SKIN DISEASE CLASSIFICATION USING CNN"

Submitted in partial fulfillment of the requirements for the award of degree of Bachelor of Engineering in

COMPUTER SCIENCE & ENGINEERING

Submitted by

ADITYA B	**1MJ17CS006**
ALLEN MATHEWS	**1MJ17CS013**
AMAL NAIR	**1MJ17CS014**
ANKITH R	**1MJ17CS019**

Under the Guidance of
MRS. R SWATHIKA
AP, Department of Computer Science &Engineering

An Autonomous Institute

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
MVJ COLLEGE OF ENGINEERING
BANGALORE-67

2020-2021

An Autonomous Institute

MVJ COLLEGE OF ENGINEERING

Whitefield, Near ITPB, Bangalore-67

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that Phase I of the Project work, entitled "**COMPUTER AIDED SKIN DISEASE CLASSIFICATION USING CNN**" is a bona fide work carried out by

ADITYA B	**1MJ17CS006**
ALLEN MATHEWS	**1MJ17CS013**
AMAL NAIR	**1MJ17CS014**
ANKITH R	**1MJ17CS019**

in partial fulfillment for the award of degree of Bachelor of Engineering in Computer Science & Engineering of the Visvesvaraya Technological University, Belagavi during the academic year 2020-21. It is certified that all the corrections/suggestions indicated for internal assessment have been incorporated in the report. The project report has been approved as it satisfies the academic requirements.

Signature of the Internal Guide	**Signature of the HOD**	**Signature of the Principal**
(MRS. R SWATHIKA)	**(MRS. R TAMILARASI)**	**(DR. P MAHABALESWARAPPA)**

An Autonomous Institute

MVJ COLLEGE OF ENGINEERING

Whitefield, Near ITPB, Bangalore-67

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We, **Aditya B, Allen Mathews, Amal Nair, Ankith R** hereby declare that the entire phase-I work of the project titled **"COMPUTER AIDED SKIN DISEASE CLASSIFICATION USING CNN"** embodied in this project report has been carried out by us during the 7th semester of BE degree at MVJCE, Bangalore under the esteemed guidance of **Mrs. R. SWATHIKA**, (AP, Dept. of CSE, MVJCE) affiliated to Visvesvaraya Technological University, Belagavi. The work embodied in this dissertation work is original and it has not been submitted in part or full for any other degree in any University.

ADITYA B	**1MJ17CS006**
ALLEN MATHEWS	**1MJ17CS013**
AMAL NAIR	**1MJ17CS014**
ANKITH R	**1MJ17CS019**

Place: MVJCE, Bangalore

Date:

ABSTRACT

There are three major types of skin cancer: basal cell carcinoma (the most common), squamous cell carcinoma (the second most common), which originate from skin cells, and melanoma. Early detection of skin cancer gives you the greatest chance for successful skin cancer treatment. Thus, we propose to use a generative adversarial network to provide us with synthetic anomalous data and augment it to the existing dataset. Most of the existing systems use a combinations of very large and complex models. These large models are favoured for their higher accuracy rates. These models sacrifice deployability for this higher level of accuracy. This means that most of these existing systems cannot be deployed as an application as they would perform badly on mobile systems like laptops or smartphones. This is makes them unfeasible for real world use in hospitals. These systems did not augment any new data that could help improve the performance of the models. More variations of the existing data can help the model be more prepared for lesions it has never seen before. Most other models used in the previous works are usually very large and can take a lot of time to process predictions and usually need a lot processing power to do so. However, such a model is rather impractical to be deployed. Our objective is to make a model which is reliable as well as lightweight enough to be deployed in real world environments.

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany a successful completion of any task would be incomplete without the mention of people who made it possible, success is the epitome of hard work and perseverance, but steadfast of all is encouraging guidance.

So, with gratitude we acknowledge all those whose guidance and encouragement served as beacon of light and crowned our effort with success.

We are thankful to our Principal **DR. P MAHABALESWARAPPA** and Vice Principal, **Prof. M BRINDHA** for their encouragement and support throughout the project work.

We are also thankful to our **HOD Mrs. R TAMILARASI** for her incessant encouragement & all the help during the project work.

We consider it a privilege and honour to express our sincere gratitude to our guide **Mrs. R SWATHIKA, Asst. Professor, Dept. of CSE** for her valuable guidance throughout the tenure of this project work, and whose support and encouragement made this work possible.

It's also a great pleasure to express our deepest gratitude to all faculty members of our department for their cooperation and constructive criticism offered, which helped us a lot during our project work.

Finally, we would like to thank all our family members and friends whose encouragement and support was invaluable.

Thanking you

Table of Contents

Abstract	**i**
Acknowledgement	**ii**
List of Figures	**v**
CHAPTER 1: INTRODUCTION	**1-4**
1.1 Project Overview	1
1.2 Purpose of Project	3
1.3 Scope of Project	3
1.4 Problem Statement	3
1.5 Existing System	3
1.6 Proposed System	3
1.7 Lesion classes in dataset	4
CHAPTER 2: LITERATURE SURVEY	**5-9**
2.1 Overview	5
2.2 Related papers	5
CHAPTER 3: SYSTEM REQUIREMENTS	**10-12**
3.1 Functional Requirement	10
3.2 Non-functional Requirement	10
3.2.1 Basic Operational Requirement	11
3.2.2 Organizational Requirement	11
3.2.3 Product Requirement	11
3.2.4 User Requirement	11
3.3 Resource Requirement	12

3.4 Hardware Requirement	12
3.5 Software Requirement	12
CHAPTER 4: SYSTEM DESIGN	**13-16**
4.1 Fundamental Design Concepts	13
4.1.1 Input Design	13
4.1.2 Output Design	13
4.2 Development of System	14
4.2.1 Different phases of model	14
4.2.2 Purpose of choosing Water fall model	15
4.3 Use case Diagram	15
4.4 Sequence Diagram	16
CHAPTER 5: IMPLEMENTATION	**17-19**
5.1 Algorithm	17
CHAPTER 6: TESTING	**20-23**
6.1 Unit Testing	20
6.2 Integration	20
6.3 Integration Testing	21
6.3.1 Up down Integration	21
6.3.2 Bottom-up Integration	21
6.4 Validation Testing	21
6.5 Testing Output	22
6.6 User Acceptance Testing:	22
6.7 Test Data Preparation	22
6.7.1 Testing Artificial Data	22

6.8 Assurance of Quality 23

6.8.1 Quality Factors 23

6.8.2 General Risks 23

6.8.3 Security Technologies & Policies 23

CHAPTER 7: RESULTS 24-28

7.1 Result Screenshot 24

7.1.1 Dataset 24

7.1.2 Training the model 25

7.1.3 Graphical User Interface 27

CHAPTER 8: CONCLUSION 29

REFERENCES 30

LIST OF FIGURES

Fig no.	Figures	Pg.no
1.1	Flowchart of Skin classification	2
2.1	Scope of Image processing and Deep learning in diagnosis of Skin Cancer	6
2.2	Schematic layout diagram of our proposed deep learning framework	7
4.1	Waterfall model for Proposed System	15
4.2	Use case diagram	15
4.3	Sequence diagram	16
7.1	Images from HAM10000 dataset	24
7.2	Metadata in HAM10000 dataset	24
7.3	Training and validation	25
7.4	Training and validation categorical accuracy	25
7.5	Training and validation top 2 accuracy	25
7.6	Training and validation top 3 accuracy	26
7.7	Confusion matrix	26
7.8	Recall and precision score	26
7.9	Graphical user interface	27

Chapter 1

INTRODUCTION

1.1 Project Overview

Skin cancer — the abnormal growth of skin cells —most skin cancers arise from DNA mutations induced by ultraviolet light affecting cells of the epidermis. Most skin cancers are locally destructive cancerous (malignant) growth of the skin. They originate from the cells of the epidermis, the superficial layer of the skin. Skin cancer is primarily diagnosed visually, beginning with an initial clinical screening and followed potentially by dermoscopic analysis, a biopsy and histopathological examination, the latter two being rather invasive procedures.

There are three major types of skin cancer: basal cell carcinoma (the most common), squamous cell carcinoma (the second most common), which originate from skin cells, and melanoma, which originates from the pigment-producing skin cells (melanocytes) but is less common, though more dangerous, than the first two classes. Early detection of skin cancer gives you the greatest chance for successful skin cancer treatment.

We clean the images by using a gaussian filter. It is widely used to reduce image noise. Image inpainting is used as one of the image restoration techniques used for the reconstruction of a lost or damaged region in an image. In this project we use inpainting to remove blemishes, hair and other undesired distortions in the images before extracting their features. The main objective of the inpainting algorithm is to replace these regions and fill up the holes in a natural way.

Like most problems in the medical domain, normal data is relatively easy to acquire however it is rather difficult to have sufficient abnormal data, especially for some rare diseases, making training a standard classifier very challenging. Thus we propose to use a generative adversarial network to provide us with synthetic anomalous data and augment it to the existing dataset. Given a training set, this technique learns to generate new data with the same statistics as the training set. Data augmentation results in better performing models, increasing both the model's skill and providing a regularizing effect, reducing generalization error. It works by creating new, artificial but plausible examples from the input problem domain on which the model is trained.

In this project, we replace traditional feature extractors by a convolutional neural network (CNN), since CNN's have a strong ability to extract complex features that express the image in much more detail, learn the task specific features and are much more efficient. Features like colour, intensity, contrast, homogeneity, texture, etc. are extracted.

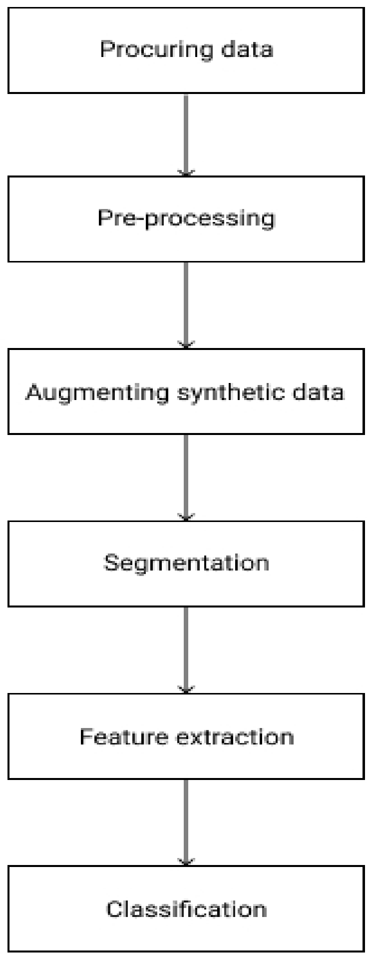

Fig 1.1: Flowchart of Skin classification

1.2 Purpose of Project

The motivation for this proposed system is that: most other models used in the previous works are usually very large and can take a lot of time to process predictions and usually need a lot processing power to do so. However, such a model is rather impractical to be deployed. Our objective is to make a model which is reliable as well as lightweight enough to be deployed in real world environments.

1.3 Scope of Project

The primary scope of project is to provide an optimal and easy to deploy detection model with good classification percentage. This can be used further in the field of science, education, medicine and many more.

1.4 Problem Statement

To develop a lightweight model to classify skin lesions which is reliable enough to be used in real world scenarios and use it to develop mobile or embedded vision applications where this model can be deployed.

1.5 Existing System

Most of the existing systems use a combinations of very large and complex models. These large models are favoured for their higher accuracy rates. These models sacrifice deployability for this higher level of accuracy. This means that most of these existing systems cannot be deployed as an application as they would perform badly on mobile systems like laptops or smartphones. This is makes them unfeasible for real world use in hospitals.These systems did not augment any new data that could help improve the performance of the models. More variations of the existing data can help the model be more prepared for lesions it has never seen before.

1.6 Proposed System

We propose to use a modified version of a pretrained model called MobileNet. MobileNet is a very lightweight model which is very suitable for smaller systems which do not have the processing power to run the commonly used large nets. We use a ImagDataGenerator to create extra data to be augmented into the existing dataset. A combination of modifications such as rotation, width shift, height shift, flipping horizantally or vertically, and zooming in or out was used to produce the new images. MobileNet has 93 layers and we train the last 23 of them. We also reduce the dropout ratio for the last 6 layers to prevent overfitting.Finally the last output layer is changed to be a 7 output dense layer. 7 outputs are used because we have 7 classes in the dataset.

1.7 Lesion classes in dataset

• **Actinic keratoses** (also called solar keratoses) are dry scaly patches of skin that have been damaged by the sun. The patches are not usually serious. But there's a small chance they could become skin cancer, so it's important to avoid further damage to your skin.

• **Basal cell carcinoma** is a type of skin cancer that most often develops on areas of skin exposed to the sun. This photograph shows a basal cell carcinoma that affects the skin on the lower eyelids.

• **Benign keratosis** skin growths that some people develop as they age. They often appear on the back or chest, but can occur on any part of the body. It grow slowly, in groups or singly. Most people will develop at least one seborrheic keratosis during their lifetime.

• **Dermatofibroma** is a common cutaneous nodule of unknown etiology that occurs more often in women. It develops on the extremities and is usually asymptomatic.

• **Melanoma** is a form of skin cancer that begins in the cells which control the pigment of your skin. Symptoms include unusual growth change of an existing mole and can occur anywhere on the body.

• **Melanocytic nevi** are benign neoplasms of melanocytes and appear in a myriad of variants, which all are included in our series. The variants may differ significantly from a dermatoscopic point of view.

• **Vascular lesions** are relatively common abnormalities of the skin and underlying tissues, most commonly birthmarks.

Chapter 2

LITERATURE SURVEY

2.1 Overview

Skin disease classification is a fairly researched topic and a lot of developments have been made in this domain. Most systems tend to use CNNs or some other similar neural networks to extract features and perform classification. Here we take a look at some of the approaches taken by contributors in the scientific field and talk about their methodology.

2.2 Related papers

Automated classification of skin lesions using images is a challenging task and it is carried out using *deep convolutional neural networks* to classify the skin lesions. Performance of the algorithm is tested using binary classification of *keratinocyte carcinomas* versus *benign seborrheic keratoses*; and *malignant melanomas* versus *benign nevi*.

This paper makes use of *vision machine technique* to remove the uncertainty of errors;
Optimal Wiener filter coefficients for noise reduction and estimating *point spread function* (PSF) for blurriness and sharpness of lesion images;area of the lesion and boundary irregularity are determined by increasing ratio of number of images of skin with malignant tumor and no. of images of skin with benign tumor identified correctly to the total no. of images of skin with malignant tumor, enhanced sensitivity using CNN; And the skin cancer detection was experimented with different category of images which was affected.

Melanoma images are taken from the DermIS (http://www.dermis.net) and DermQuest http://www.dermquest.com) database.This paper focuses on detecting skin cancer, based on specific features such as the area of the spread, diameter, color of the lesion, etc. Skin lesion test is performed to evaluate the performance of the proposed method. The test data have revealed that CNN out performs when the features like average *High Pass Filter*(HPF) and *point spread function (*PSF) were included for training the model.

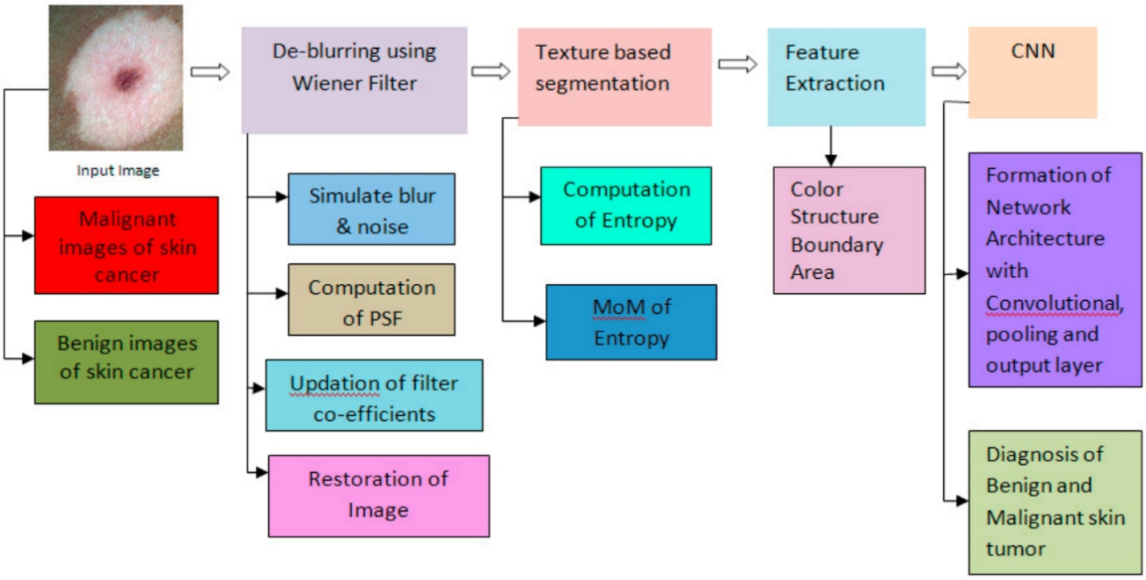

Fig 2.1 Scope of Image processing and Deep learning in diagnosis of Skin cancer

While the **problems** in this paper faced were power spectra of the random fields to which picture and noise are assumed to belong must be known or estimated in Wiener filter; No accurate accuracy of all components in PSF if the sampling rate of the star image is too low; HPF is sensitive in detuning of fundamental frequency and parameter components of images that are taken to classify.

The estimated deviation in sensitivity is in the range of **6.7% to 14.3%** and it effectively improved the identification efficiency of the integrated machine vision melanoma detection system.

In this paper the ISIC 2019 challenge dataset has been developed here by using transfer learning and a pre-trained deep neural network *GoogleNet.*

The proposed model successfully classified the **eight** different classes of skin lesions accurately including the imbalance of images between classes namely, *Melanoma*, *Melanocytic nevus*, *Basal cell carcinoma*, *Actinic Keratosis*, *Benign Keratosis*, *Dermatofibroma*, *Vascular lesion*, and *Squamous cell carcinoma.*

The performance of the proposed method by this author increased when the number of images in all classes decreased to overcome the problem of imbalance in images between classes.When the weights of all architecture layers were fine tuned, the performance measures become higher than fine tuning only the replaced layers.

The ISIC 2019 *dataset* contains images of *HAM10000* and the *BCN_20000*. HAM10000 contains ~10000 images with a size of 600×450. This dataset was the older challenge of ISIC 2018. The BCN_20000 contains ~19424 images of size 1024×1024.
Only 10% of the dataset is considered training,validation and testing. The performance measures of the proposed methods were *Accuracy*=94.92%, *Sensitivity*=79.8%, *Specificity*=97%, and *Precision*=80.36%.

The **problems** that were faced are, can't be trained or tested using the same device; it requires a specific hardware specification that all researchers in various countries cannot meet, only works if the initial and target problems of both models are similar, and *overfitting* happens when the new model learns details and noises from training data that negatively impact its outputs.

This paper proposes a new framework that performs both segmentation and classification of skin lesions for automated detection of skin cancer. It comprises of 2 stages:

- ***Encoder-Decoder Fully Convolutional Network*** **(FCN)**

  to learn the complex and inhomogeneous skin lesion features with the encoder stage learning the coarse appearance and the decoder learning lesion borders details.
- ***Novel FCN-based DenseNet framework***

  to compose dense blocks that are merged and connected via the concatenation strategy and transition layer

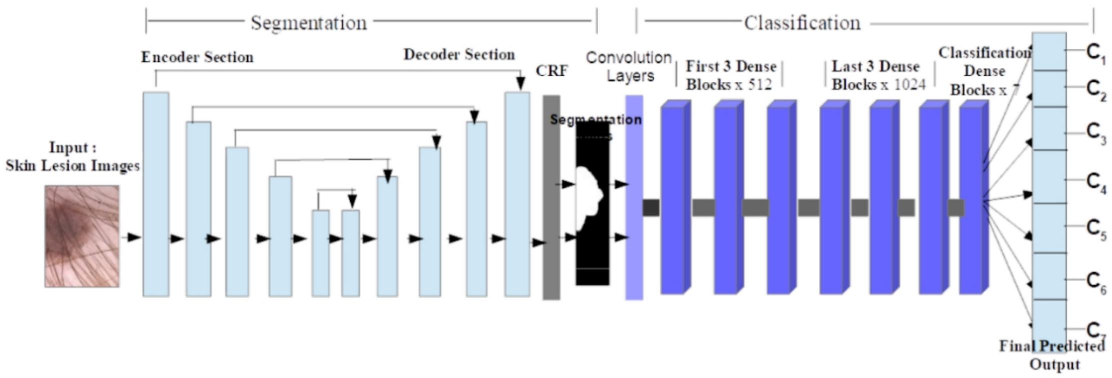

Fig 2.2 Schematic layout diagram of our proposed deep learning framework for skin lesion images segmentation and classification

The classification system was evaluated separately on unsegmented images to show the effect of the segmentation network which made it comparatively slower.In this paper, the proposed system has been able to overcome the challenges of dealing with the complex features of skin lesion images and heavy parameter tuning of the traditional CNN.

The system employs optimization techniques to reduce network complexity and improve computing efficiency, encourages feature reuse and thus requires a small number of parameters and is effective with limited data.Their training data contains ~10030 images and 1 ground truth response CSV file was taken from *HAM10000*

The performance measures of the proposed methods were *Accuracy*=98%, *Recall*=98.5%, and *Specificity*=99%. While the **problems** that were faced by these authors were, as there are (L(L − 1)/ 2) connections for a L-layer DenseNet. The excessive connections not only decrease networks' computation-efficiency and parameter-efficiency, but also make networks more prone to *overfitting,* DenseNet uses a *lot more memory.*
The statistic behind the rise in Skin Cancer can be mainly credited to the UV rays which depletes the ozone layer. The conventional approach to detect and identify a particular type of cancer can be agonizing for the patient as skin tissues are to be extracted from their body for analysis. One of the most common digital tools used is dermoscopy as they are less expensive and have high accuracy as they have a massive source of datasets. But there is still a need to implement a tool to help small medical practices and dermatologists provide a diagnostic analysis. With the help of CNN (Convolutional Neural Network), Image Processing can be implemented to detect early stages of skin cancer. There may be some instances where the image quality may be distorted or noisy. This can be enhanced using CLAHE (Contrast Limited Adaptive Histogram Equalization) and MSRCR (MultiScale Retinex Color Restoration). For a set of dataset images, 75% of them are used for training the model and 25% is used for validating the model. They are enhanced using CLAHE and MSRCR as shown in the figure. A CNN classifier is used for this implementation as it provides high accuracy and efficiency. The classifier runs 2 types of image. The original image and the image enhanced by CLAHE and MSRCR. The validation accuracy has an average value of 0.85 for original dataset and CLAHE dataset and is greater than MSRCR datasets which has the value 0.8161. Therefore CLAHE is implemented for images with low contrast and MSRCR is implemented for images with minimal visibility conditions.
Diagnosis of cancer using surgical methods mainly depends on the tumor thickness. Determining the cancer stage and treatment in early stages increases the survival rate. The proposed paper uses

Convolutional Neural Network (CNN) with Similarity Measure for Text Processing (SMTP) as loss function to determine the cancer stage.

The algorithm classifies melanoma cancer stages based on tumor thickness without involving any surgical methods. There are two classification system proposed,

1. It classifies as melanoma as stage 1 and stage 2 if their tumor thickness is <0.76 mm and >=0.76 mm respectively.
2. It classifies melanoma as stage 1, stage 2 and stage 3 if their tumor thickness is <0.76 mm, >=0.76 and <1.5 and, >1.5 respectively.

The proposed algorithm CNN with SMTP is built with the following architecture:
Input consists of n entries: x_1, $x_2 \ldots x_n$, where n represents total features in the dataset and x represents the feature.
Convolutional Operation**,** it works as the feature detector and helps with the neural network's filter. Rectified Linear Unit (ReLU), It is used as the activation function. Pooling, summarizes the features present in a region of the feature map generated by a convolution layer.Full Connection**,** this is the one of the last layers of the network where output from the final Pooling or Convolutional Layer is fed as the input and merged together. Softmax, it is the output layer and where the vector of probabilities is used specify the class of every input. Loss Function (SMTP), The loss function calculates the difference between real value and predicted value. Dataset consists of 250 images of melanoma cancer consisting of 81 extracted features or attributes.

In the field of medicine, doctors and artificial intelligence compete with each other to improve on medical diagnosis of diseases. However, there is no data related to skin cancer detection that combines human and artificial intelligence. The paper proposes the benefits of combining human and artificial intelligence for skin cancer classification. Here, both physicians and a convolutional neural network (CNN) are allowed to classify dermoscopic images of skin lesions into five diagnostic categories in skin cancer screening setting independently. The decisions obtained were merged to form an overall classifier which would take the degree of certainty of both systems into account. If the error of the two systems were weakly correlated, a higher diagnostic accuracy was expected to be achieved.

Chapter 3

SYSTEM REQUIREMENTS

The system requirements specify features, components and behaviour of system which is to be developed. The following sections describe about functional, non-functional, performance related, features and behaviour of the solution. This includes the detailed description of the solution to be developed.

3.1 Functional Requirement

Functional Requirements are those requirements which show the working and functionality of a system and the expected behaviour of a system based on certain situations and inputs. It defines specific functionality of a system. The functionalities to be implemented are :-

1. Take any skin lesion image for classification.
2. Resize the image to be passed to the model.
3. Perform predictions on the processed image using the model.
4. Produce the results and observe the accuracy.

3.2 Non-functional Requirement

Non-functional requirements is not about functionality or behaviour of system, but rather are used to specify the capacity of a system. They are more related to properties of system such as quality, reliability and quick response time. Non- functional requirements come up via customer needs, because of budget, interoperability need such as software and hardware requirement, organizational policies or due to some external factors such as:-

1. Product Requirements
2. Organizational Requirements
3. User Requirements
4. Basic Operational Requirements

3.2.1 Basic Operational Requirement

The eight primary functions of systems engineering are all performed by the end users, which is the customers. Operational requirements which are given by:-

• Mission profile or situation: The mission of the project is efficient skin cancer classification system.

• Performance: It basically gives system parameters to reach our goal. Parameters for the proposed system are less computational time while retaining a respectable level of accuracy which is compared to the existing system.

3.2.2 Organizational Requirement

The Organizational requirement consists of the following types:

• Process Standards: To make sure the system is a quality product, IEEE standards have been used during system development.

• Design Methods: Design is an important step, on which all other steps in the engineering process are based on. It takes the project from a theoretical idea to an actual product. It gives us the basis of our solution. Because all the steps after designing are based on the design itself, this step affects the quality of the product and is a major player in how the testing and maintenance of a project take place and how successful they are.

3.2.3 Product Requirement

• Portability: The project can work on any platform

• Accuracy: The detection accuracy must be high.

• Correctness: The system has been put through rigorous testing after it has followed strict guidelines and rules. The testing has validated the data.

• Ease of Use: The user interface allows the user to interact with the system at a very comfortable level with no hassles.

• Modularity: The many different modules in the system are neatly defined for ease of use and to make the product as flexible as possible with different permutations and combinations.

• Robustness: During the development of the system special care is being taken to make sure that the end results are optimized to the highest level and the results are relevant and validated.

3.2.4 User Requirement

• The user must be able to view the results and understand what the results mean.

• The GUI must be easy to use and informative.

3.3 Resource Requirement

The project will be developed in python with IDLE python 3.8 environment and the GUI will be hosted using node and accessed through web browser.

3.4 Hardware Requirement

The following is the hardware requirements of the system for the proposed system:

- Processor : 2.2 GHz
- RAM : 16GB
- Hard Disk : 100 GB
- Display : 15inch colour LCD

3.5 Software Requirement

The following is the software requirements of the system for the proposed system:

- OS : Windows 10 or above
- Platform : Python 3.8
- Language : Python
- IDE/tool : IDLE
- Packages : tensorflow 2.5, keras, matplotlib, pandas, numpy, tensorflow.js

CHAPTER 4

SYSTEM DESIGN

Design is an innovative procedure; a successful system is developed through great designs. System design is a process of giving detailed information about the proposed work in a physical format. Different designs are built for development of system, which describes about features, components which are included and how client interact with system.

4.1 Fundamental Design Concepts

Fundamental design is developed in course of recent years. As year's passes, enthusiasm of creating new designs is evolved and each design has been tested. Software designer gets new ideas and foundation to build and test new design concepts. Fundamental framework is design to "getting it right". Major plan ideas, for example, deliberation, refinement, modularity, programming engineering and data encryption is applied to meet the requirement of proposed work.

4.1.1 Input Design

Input Design is a way toward changing user-based inputs into computerized format. Main objective of input design is, to make computerization as possible and error free. Giving a decent information configuration to the application simple information and determination highlights are received. The input design prerequisites, for example, ease of use, reliable organization and intelligent exchange to help client to get proper information on time. Input design is a general framework which exceptionally cautious consideration. Gathering all input parts is one of the costly parts of framework.

4.1.2 Output Design

Output design meets the necessities of users and presents the output data clearly. In any framework processing result are conveyed to clients and different frameworks in form of output design. It is direct source to client. Productive output enhances framework association with machines of source and destination. Output is an optimized cluster which is get for the proposed system. The optimized clusters also include number of centroids which is calculated for each clusters, execution time of GSO as well as graph for time taken to execute the amount of data specified.

4.2 Development of System

System development method is a process through which a product will get completed or a product gets rid from any problem. Software development process is described as a number of phases, procedures and steps that gives the complete software. It follows series of steps which is used for product progress. The development method followed in this project is waterfall model.

4.2.1 Different phases of model

The waterfall model is a successive programming improvement process, in which advance is seen as streaming relentlessly downwards (like a waterfall) through the periods of Requirement Analysis, Design, Implementation, Testing and upkeep.

- **Requirement:** This phase includes collection of all requirements which is needed for development of software.
- **Design:** The specification of system is converted into software design, by keeping in mind system specification. The designer mainly describes about algorithm, architecture and structure of system.
- **Coding:** Developer begins coding with a specific end goal to give a full outline of project. As such framework system specification are just changed over into system decipherable process code.
- **Implementation:** The execution stage includes the genuine coding or programming of the product. The yield of this stage is regularly the library, executables, client manuals and extra programming documentation
- **Testing:** The testing stage includes all modules of project which is coordinated and tried to guarantee that total framework meets product necessities. Verification and Validation is mostly concerned under this testing phase.
- **Maintenance:** The upkeep stage is the longest stage in which the product is upgraded to satisfy the changing client need, adjust to suit change in the outside environment, right mistakes and oversights beforehand undetected in the testing stage, improve the proficiency of the product.

4.2.2 Purpose of choosing Water fall model

- Objectives of project should be clear.
- Understandable software requirement.
- System progress should be measurable.
- Allocation of resources should be better.
- Guides to be prefect.
- Stable project requirements.
- Quality of project is improved.

The following fig gives the waterfall model for proposed system. The requirement of our proposed project is:

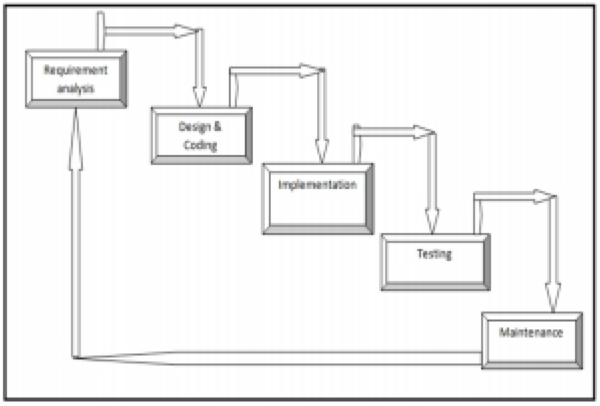

Fig 4.1:- Waterfall model for Proposed System

4.3 Use case Diagram

A use case diagram is a type of behavioral diagram created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases.

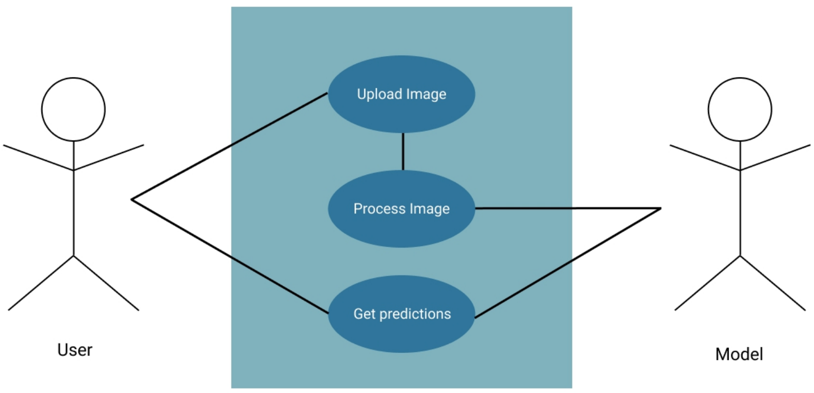

Fig 4.2: Use Case Diagram

4.4 Sequence Diagram

Sequence diagram is sort of interaction diagram which describe about process of work done to execute project.

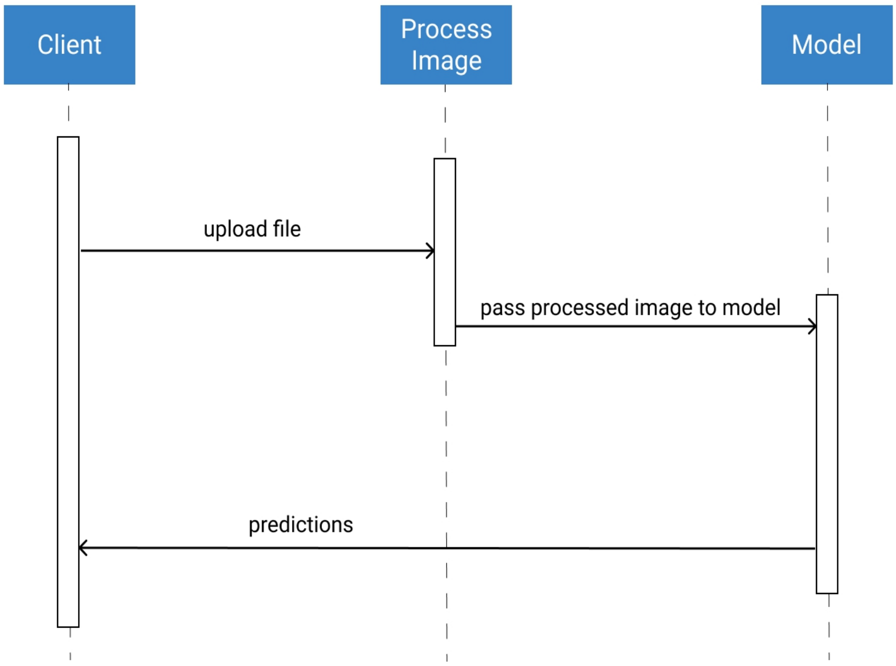

Fig 4.3 Sequence diagram of the class

Chapter 5

IMPLEMENTATION

It is main part of project where project is implemented. Implementation must be clearly defined, planned carefully and systematically otherwise it causes confusion and leads to generation of problems. Following tasks involved in stage of implementation:

- Planning carefully
- System Investigation.
- Investigating different types of systems and constraints.
- Selecting the most accurate and correct language for developing the application.
- Understanding and correctly evaluating several changeover methods.
- Rightful decision is made about the selection of the platform.

The implementation details of the project are given in detail in this section.

5.1 Algorithm

Following are the important stages in the algorithm

1. Procure the dataset and separate the images to folders corresponding to their classes. Also separate validation and training data into different folders.
2. Iterate through all classes and generate slightly modified images to be augmented to the existing data.
3. Resize the images to 224 x 224 pixels.
4. Load the MobileNet model and set the last 23 layers as trainable. Also set the dropuot ratio for the final 6 layers to 0.25 and make a new 7-Dense layer as the output.
5. Create a new Model with the inputs being mobile.input and the output being the new 7-Dense layer.
6. Define the class weights and set the class weights for melanoma higher than the rest to make the model sensitive to melanoma.
7. Train the model with the augmented data.
8. Evaluate the model with the validation data and produce the result.

We procure the HAM10000 skin lesion dataset. The dataset consists of just over 10000 images stored in two folders and the metadata is stored in a *.csv* file. We remove the duplicate images from the dataset. We use the python os library to make the new directories and then store the images into their respective folders by comparing the file names and corresponding classes from metadata file. Then we split the validation images and store them into their separate folders. Once all the images are separated, we use a data generator called *ImageDataGenerator.*

Data Augmentation is a technique of creating new data from existing data by applying some transformations such as flips, rotate at a various angle, shifts, zooms and many more. Training the neural network on more data leads to achieving higher accuracy. In real-world problem, we may have limited data. Therefore, data augmentation is often used to increase train dataset. It is a keras deep learning library provides the data augmentation function, which applies augmentation automatically while training the model. An ImageDataGenerator class function provides a range of transformations such as:

- Translations
- Rotations
- Shearing
- Changes in scale
- Image flipping
- Zooming

We used the following parameters to generate the images for augmentation:

- **rotation_range**: Integer value describing the degree range for random rotations. In our implementation it was 180 degrees.
- **width_shift_range**: Float value, fraction of total width, if < 1, or pixels if >= 1. We specified this range to 0.1 (10 %).
- **height_shift_range**: Float value, fraction of total height, if < 1, or pixels if >= 1. We specified this range to 0.1 (10 %).
- **zoom_range**: Float value or [lower, upper]. Range for random zoom. If a float, [lower, upper] = [1-zoom_range, 1+zoom_range]. We specified this value as 0.1 or 10%.
- **fill_mode**: One of {"constant", "nearest", "reflect" or "wrap"}. Default is 'nearest'. Points outside the boundaries of the input are filled according to the given mode. We used '*nearest'* fill mode.

- **horizontal_flip**: Boolean. Randomly flip inputs horizontally. We set this value as True.
- **vertical_flip**: Boolean. Randomly flip inputs vertically. We set this value as True.

Once the augmentation is complete, we resize all the images using numpy to 224 x 224px. The original size of the images is 600 x 450px. This resizing allows to improve the training speed while sacrificing the least amount of performance.

After all the images are processed, we import the MobileNet model from keras. MobileNet is a pretrained model. MobileNets are based on a streamlined architecture that uses depth-wise separable convolutions to build light weight deep neural networks. There are a total of 93 layers in the model. We set the last 23 layers as trainable.

The dropout rate of the last 6 layers is also reduced from 1.0 to 0.25. The final output layers are modified to be a dense layer with 7 outputs. The 7 layers corresponding to the number of classes in the problem. Then we create a new model using *keras.Model* where the inputs are the MobileNet inputs an the outputs are the new 7-Dense layer as the output.

This model is compiled with the *Adam* optimizer and the loss function as *CategoricalCrossentropy.* The purpose of loss functions is to compute the quantity that a model should seek to minimize during training. When training a neural network, its weights are initially initialized randomly and then they are updated in each epoch in a manner such that they increase the overall accuracy of the network. In each epoch, the output of the training data is compared to actual data with the help of the loss function to calculate the error and then the weight is updated accordingly. This is essentially an optimization problem where the goal is to optimize the loss function and arrive at ideal weights. The method used for optimization is known as *Optimizer*.

We define two functions, top_2_accuracy and top_3_accuracy. These are used to observe the accuracy and other metrics of the top 2 predictions and the top 3 predictions respectively. The model is then trained using our dataset.

After the training is finished, the model is evaluated against the validation data. The results are plotted and observed.

Chapter 6

TESTING

Testing is an important part of project where testing of each module is done. Testing guarantees that proposed system is well organized analyzed to meet require project goal. Testing is last stage of project which guarantees the system is error free and ready to give desired output. The goals of testing are given by:

- Give operational quality to system
- Search and remove errors.
- Best quality project is produced.
- To approve the product as a solution for the first issue.

The following are types of testing performed in proposed system.

6.1 Unit Testing

Each module of project is tested individually. Verification is done on each module. Module of project is tested individually. Testing is done in programming style. The unit testing for the proposed system is performed on initialization, centroid selection, sub-coverage distance and clustering movement.

6.2 Integration

After unit testing is performed, integration testing takes place in each module of project. Integration is done on various classes of system. It is done on front and back end also.

➢ Function into classes Integration

Initially during code phase various functions is developed for development of system.Each function of system is tested and coded individually. As all the functions are verified they are mixed into their particular classes.

➢ Distinct classes Integration

Based on functionality, testing of distinct classes is done independently. Verification of each class is done which gives good result and hence integration is performed again on different classes.

➢ Front & Back end Integration

Swing java environment builds front end of project. Development of User Interface (UI) facilitates user to work on various input commands and get desired output of system. Hence backend is integrated with GUI and later tested.

6.3 Integration Testing

Developing a programming framework is a sophisticated technique which is used by Integration testing. It solves various issues on dual verification problem and construct program which solves all related problems. Main objective of integration testing is to construct a program structure based on unit testing modules.

As modules of software are divided, testing is performed on each module. Later this separated module is tested as whole set. Here, to rectify errors is difficult as it has different isolated errors.

6.3.1 Up down Integration

The up down integration deals with development of program framework in incremental way. Modules of each program are co-ordinated in descending order and it starts with primary module.

6.3.2. Bottom-up Integration

This method begins the construction and testing with the modules at the lowest level in the program structure. Since the modules are integrated from bottom to up, processing required for modules subordinate to a given level is always available. Therefore in this case the need for stubs is eliminated.

6.4 Validation Testing

As completion of testing combination is done, writing computer program is put together in on package. Testing approval is described from various perspectives. Hence testing affirms items limits which are sensibly expected.

6.5 Testing Output:

After performing the validation testing, the next step is output testing of the proposed system, since no system could be useful if it does not produce the required output in the specified format. Therefore the output testing involves first of all asking the users about the format required by them and then to test the output generated or displayed by the system under consideration. The output format is considered in 2 ways: –

1. On screen
2. Printed format

Printed format is used for validating the sites crawled, the relevance score of sites, the ranking score of sites.

6.6 User Acceptance Testing:

Key factor for success of any system is user acceptance. User acceptance testing is performed on users, show which will be successful based on user motivation and knowledge. At time of developing and making required changes, system under consideration along with prospective system, users undergoes constant testing. The changes are made regarding to 3 points

- Input design
- Output design
- Menu driven system

6.7 Test Data Preparation

In system testing, test data preparation plays a main role. Test data is prepared and is used to test the system under study where errors are hidden and removes by using following testing steps for better future use.

6.7.1 Testing Artificial Data

Artificial data is created for testing purpose. All combinations of formats and values are tested by generating the artificial test data. Data generating utility program in system information departments helps to prepare artificial data as quickly as possible.

6.8 Assurance of Quality

Quality assurance is testing and analysis of administration element. The main objective is to give knowledge about item quality to administration. Assurance of quality involves:

6.8.1 Quality Factors

The main goal of confirmation value is track product quality and observes procedures to enhance programming. Quality factors mainly focus on following these things:

- Operational attributes
- Experiences capacity changes
- Versatility.
- Effectiveness
- Time duration.

6.8.2 General Risks

Risk is nothing but which gives negative results at undesirable incident. The following three things are considered to recognize risk in other projects:

- Damage occurs during an occasion.
- Probability of occurring an occasion.
- The result that modified at certain level.

6.8.3 Security Technologies & Policies

The seven major activities the software quality is comprised of which are follows:

- Conduct of formal specialized audits
- Measurement
- Testing Software
- Application for specialized techniques.
- Control of progress
- Record keeping and announcing
- Enforcement of measures

CHAPTER 7

RESULTS

7.1 Result Screenshot

7.1.1 Dataset

The Dataset for the following model requires two documents. One document represents the suspicious document and second document represents the source document.

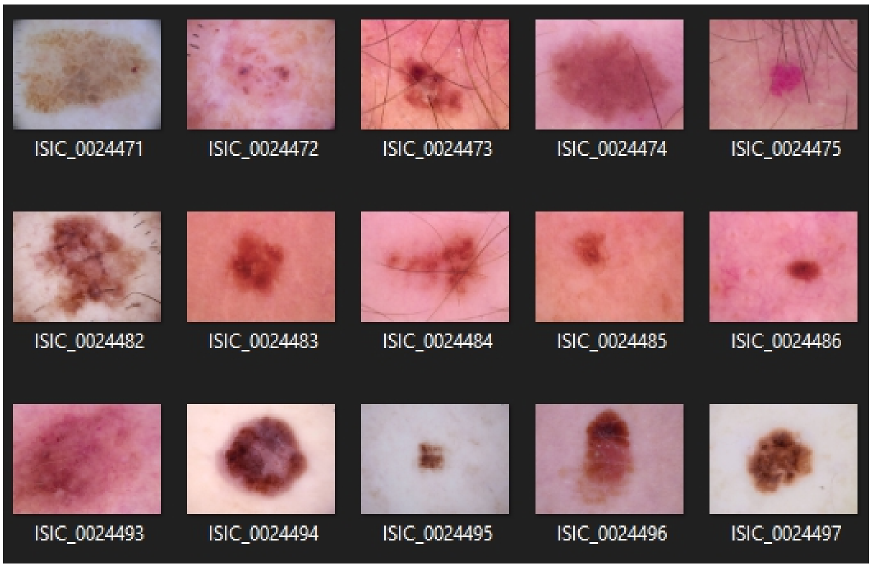

Fig 7.1: Some images from the HAM10000 dataset

HAM_0001949	ISIC_0032417	bkl	histo	70	male	trunk
HAM_0007207	ISIC_0031326	bkl	histo	65	male	back
HAM_0001601	ISIC_0025915	bkl	histo	75	male	upper extremity
HAM_0001601	ISIC_0031029	bkl	histo	75	male	upper extremity
HAM_0007571	ISIC_0029836	bkl	histo	70	male	chest
HAM_0007571	ISIC_0032129	bkl	histo	70	male	chest
HAM_0006071	ISIC_0032343	bkl	histo	70	female	face
HAM_0003301	ISIC_0025033	bkl	histo	60	male	back
HAM_0003301	ISIC_0027310	bkl	histo	60	male	back
HAM_0004884	ISIC_0032128	bkl	histo	75	male	upper extremity
HAM_0004884	ISIC_0025937	bkl	histo	75	male	upper extremity
HAM_0002521	ISIC_0027828	bkl	histo	40	male	upper extremity
HAM_0002521	ISIC_0029291	bkl	histo	40	male	upper extremity
HAM_0006574	ISIC_0030698	bkl	histo	40	male	back
HAM_0006574	ISIC_0025567	bkl	histo	40	male	back
HAM_0001480	ISIC_0031753	bkl	histo	70	male	abdomen
HAM_0001480	ISIC_0026835	bkl	histo	70	male	abdomen

Fig 7.2: Metadata in the HAM10000 dataset

7.1.2 Training the model

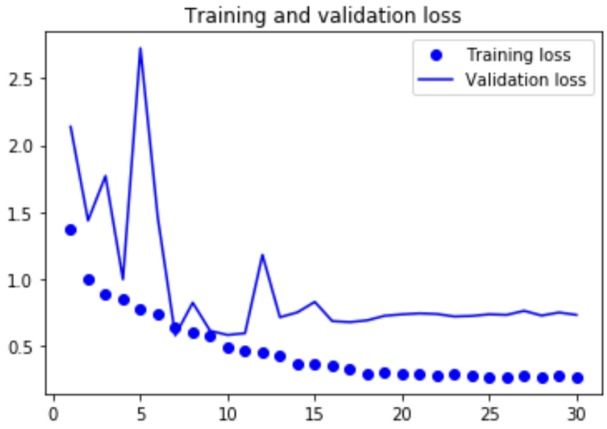

Fig 7.3: Training and validation

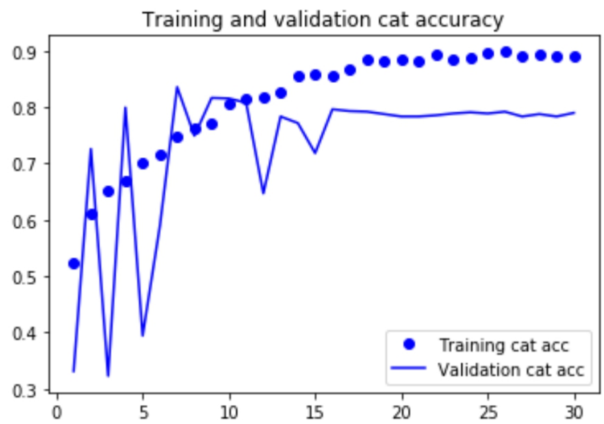

Fig 7.4: Training and validation categorical accuracy

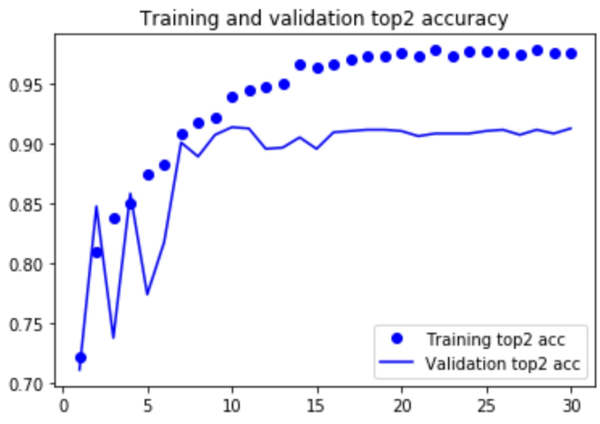

Fig 7.5: Training and validation top 2 accuracy

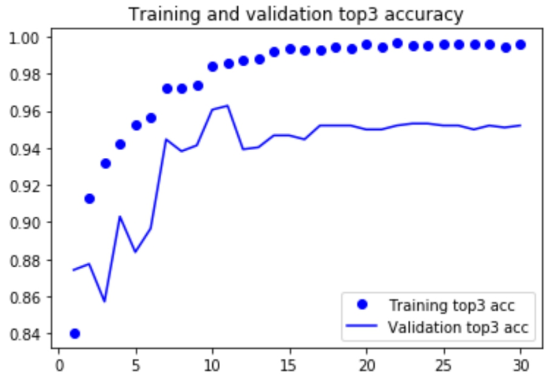

Fig 7.6: Training and validation top 3 accuracy

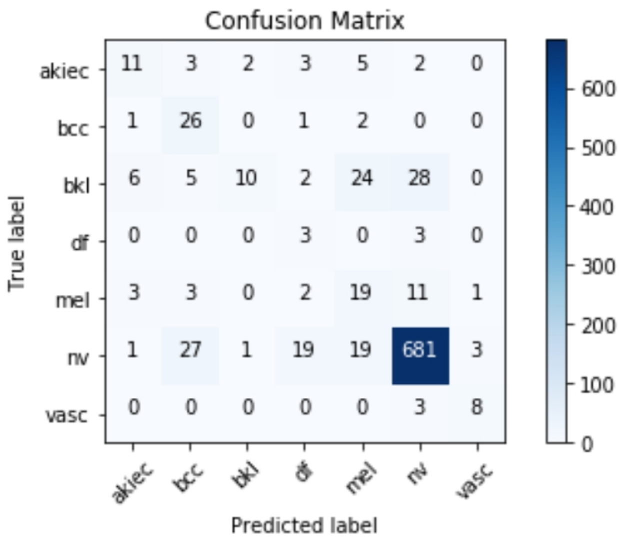

Fig 7.7: Confusion matrix

	precision	recall	f1-score	support
akiec	0.50	0.42	0.46	26
bcc	0.41	0.87	0.55	30
bkl	0.77	0.13	0.23	75
df	0.10	0.50	0.17	6
mel	0.28	0.49	0.35	39
nv	0.94	0.91	0.92	751
vasc	0.67	0.73	0.70	11
avg / total	0.86	0.81	0.81	938

Fig 7.8: Recall and precision score

7.1.3 Graphical User Interface

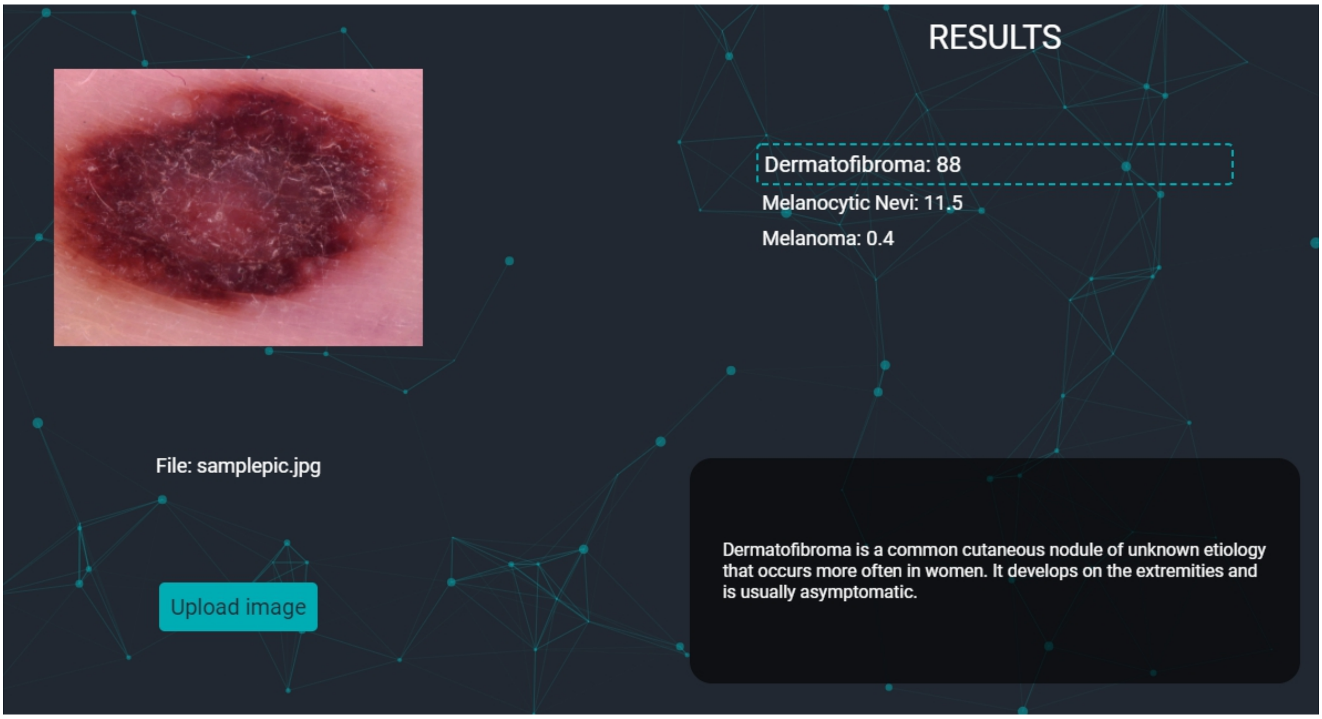

Fig 7.9: Graphical user interface (a)

Fig 7.9: Graphical user interface (b)

Fig 7.9: Graphical user interface (c)

CHAPTER 8

CONCLUSION

This model classifies skin lesions into seven classes. It is a fine tuned MobileNet CNN. The main challenges were the unbalanced dataset and the small amount of data. We used data augmentation to reduce the class imbalance and in so doing get categorical accuracy scores that were not heavily skewed by a single majority class. MobileNet's small size and speed makes it ideal for web deployment. Tensorflow.js is a new library that allows machine learning models to run in the browser - without having to download or install any additional software. Because the model is running locally, any data that a user submits never leaves his or her PC or even their mobile phone. We imagine that privacy is especially important when it comes to medical data. We created an online tool that can tell doctors and lab technologists the three highest probability diagnoses for a given skin lesion. This will help them quickly identify high priority patients and speed up their workflow. The app should produce a result in less than 3 seconds. To ensure privacy the images are pre-processed and analyzed locally and never uploaded to an external server. More advancements and developments can be made in this field in the coming years.

REFERENCES

[1] M. A. Kassem, K. M. Hosny and M. M. Fouad, "Skin Lesions Classification Into Eight Classes for ISIC 2019 Using Deep Convolutional Neural Network and Transfer Learning," in IEEE Access, vol. 8, pp. 114822-114832, 2020, doi: 10.1109/ACCESS.2020.3003890.

[2] Adegun, Adekanmi A. and S. Viriri. “FCN-Based DenseNet Framework for Automated Detection and Classification of Skin Lesions in Dermoscopy Images.” *IEEE Access* 8 (2020): 150377-150396.

[3] Sudhriti Sengupta, Neetu Mittal & Megha Modi (2019): Improved Skin Lesions Detection Using Color Space and Artificial Intelligence Techniques, Journal of Dermatological Treatment, DOI: 10.1080/09546634.2019.1708239

[4] Liu L, Mou L, Zhu XX, Mandal M. Automatic skin lesion classification based on mid-level feature learning. Comput Med Imaging Graph. 2020 Sep;84:101765. doi: 10.1016/j.compmedimag.2020.101765. Epub 2020 Jul 21. PMID: 32810817.

[5] Hameed, N., Shabut, A. M., Ghosh, M. K., & Hossain, M. A. (2020). Multi-class multi-level classification algorithm for skin lesions classification using machine learning techniques. *Expert Systems with Applications*, *141*, [112961]. https://doi.org/10.1016/j.eswa.2019.112961

[6] Hekler A, Utikal JS, Enk AH, Hauschild A, Weichenthal M, Maron RC, Berking C, Haferkamp S, Klode J, Schadendorf D, Schilling B, Holland-Letz T, Izar B, von Kalle C, Fröhling S, Brinker TJ; Collaborators. Superior skin cancer classification by the combination of human and artificial intelligence. Eur J Cancer. 2019 Oct;120:114-121. doi: 10.1016/j.ejca.2019.07.019. Epub 2019 Sep 10. PMID: 31518967.

[7] https://arxiv.org/abs/1803.10417v2