SOLENT UNIVERSITY FACULTY OF BUSINESS LAW AND DIGITAL TECHNOLOGIES DEPARTMENT OF SCIENCE AND ENGINEERING

MSc Applied AI and Data Science

Academic Year 2023-2024

Student Name

ALBERT PULICKEEL MICHAEL

Student id : Q16441494

Title of the Dissertation

SKIN DISEASE CLASSIFICATION SYSTEM

SUPERVISOR: ZUBAID KHAN

JANUARY 2024

This report is submitted in partial fulfilment of the requirements of Solent University for the degree of MSc Artificial Intelligence and Data Science.

# ABSTRACT

In the field of artificial intelligence and advanced data science, this research aims at the convergence of computer vision, deep learning and medical diagnosis. Focusing on the well-known architecture of VGG-16, a fundamental mystery is explored: the exact impact of image quality on the performance of skin disease classification models. This thesis addresses the challenges posed by images subject to variable compression, where file sizes fluctuate, which can affect image quality.it is Motivated by real-world scenarios where image quality is rarely consistent, this study attempts to decipher how the VGG-16 model handles different image qualities. Ethical considerations are paramount and steps will be taken to ensure responsible and transparent use of data, minimize bias and ensure research integrity. The thesis is presented in chapters, culminating in a proposed solution to address the challenges posed by different image qualities in the implementation of image classification systems.

The original skin classification system uses the ISIC dataset, which collects images of diseased and healthy skin to train a convolutional neural network (CNN) model, specifically the VGG-16 architecture. The model has undergone parameter tuning to ensure its accuracy and usefulness. The developed web application allows users to upload skin images for classification purposes. The user-friendly interface has a dedicated prediction button to start the classification process. To evaluate the robustness of the system, different qualities of the same image were downloaded and the resulting predictions recorded. This evaluation aims to understand how a web application's prediction may change based on different image quality. Integrating the VGG-16 model into the web application provides a transparent and accessible means of skin classification for users.

# 

# ACKOWLEDGEMENT

I want to express my sincere gratitude towards Solent University for giving me the chance to do this research in Applied AI and Data Science. Although challenging, the journey of this dissertation has been enlightening, and I appreciate the institution's continued guidance and solid support.

I extend sincere thanks to my supervisor, Zubaid Khan, whose insights, patience, and dedication significantly influenced my research. Throughout this project, your encouragement and constructive criticism have been invaluable.

Thanks also to the all faculty and staff of the Department of Science and Engineering for fostering an environment of academic excellence. The impact of this environment on my academic journey at Solent University has been truly worthwhile.

I'm grateful to my colleagues who shared this academic journey. Thank you for collaborative learning, late-night brainstorming sessions, and the camaraderie that made even the toughest days bearable.

Lastly, I convey heartfelt gratitude to my family. Your unwavering support, belief in my capabilities, continued encouragement, and boundless love have been a source of immense strength throughout this journey. Without your unwavering support, this achievement would not have been possible.

In conclusion, while my name is on this dissertation, it is the result of the collective efforts, guidance, and support of many. To everyone who has been part of this journey, directly or indirectly, I express my sincerest gratitude.

Top of Form

Contents

[ABSTRACT 3](#_Toc155621174)

[ACKOWLEDGEMENT 3](#_Toc155621175)

[LIST OF TABLES 7](#_Toc155621176)

[ACRONYMS 7](#_Toc155621177)

[Chapter 1 8](#_Toc155621178)

[Introduction 8](#_Toc155621179)

[1.1 Background 8](#_Toc155621180)

[1.2. Problem Context 11](#_Toc155621181)

[1.3 Purpose 11](#_Toc155621182)

[1.4 Proposed solution 11](#_Toc155621183)

[1.5 Aim, Objectives and Scopes 12](#_Toc155621184)

[Aim 12](#_Toc155621185)

[1.6 ETHICAL CONSIDERATION 12](#_Toc155621186)

[1.7 DESERTATION OUTLINE 13](#_Toc155621187)

[Chapter 2 13](#_Toc155621188)

[LITERATURE REVIEW 13](#_Toc155621189)

[2.1 Topic 1: Automated Systems for Skin Disease Detection 13](#_Toc155621190)

[2.1.1 Subtopic 1: Mobile Applications for Skin Disease Diagnosis 13](#_Toc155621191)

[2.1.2 Deep Learning Models for Skin Disease Detection 14](#_Toc155621192)

[2.1.3 Symptom based classification of Skin Diseases 15](#_Toc155621193)

[2.2 WEB-BASED SYSTEMS FOR SKIN DISEASE DETECTION 16](#_Toc155621194)

[2.2.1 ADDRESSING HEALTHCARE GAPS THROUGH WEB-BASED SYSTEMS 16](#_Toc155621195)

[2.2.2 RULE-BASED SYSTEMS FOR SKIN DISEASE RECOGNITION 17](#_Toc155621196)

[2.2.3 Diagnosing skin diseases on mobile devices: Integrating image processing, data mining and user interaction 17](#_Toc155621197)

[2.2.4 TRANSFER LEARNING WITH VGG-16 for SKIN DISEASE CLASSIFICATION 18](#_Toc155621198)

[Summary and Implications 20](#_Toc155621199)

[Chapter 3 20](#_Toc155621200)

[3.1 METHODOLOGY 20](#_Toc155621201)

[3.1.1 Collecting the data 20](#_Toc155621202)

[3.1.2 Pre-processing of collected images. 21](#_Toc155621203)

[Image data generator 21](#_Toc155621204)

[3.1.3.1 MODEL ARCHITECTURE 21](#_Toc155621205)

[VGG-16 ARCHITECTURE 22](#_Toc155621206)

[3.1.2 MODEL DEVELOPMENT 23](#_Toc155621207)

[3.1.2.1 VGG16 MODEL 23](#_Toc155621208)

[3.1.2.2 VGG16 Function 23](#_Toc155621209)

[3.1.2.3Model Development Work Flow 24](#_Toc155621210)

[3.1.3 MODEL EVALUATION 25](#_Toc155621211)

[3.1.4 Hyper parameter Tuning 25](#_Toc155621212)

[3.1.3 Classification of images using web app. 27](#_Toc155621213)

[3.1.7 UPLOADING TO THE WEB APP 28](#_Toc155621214)

[3.1.8 CHECKING AND MARKING RESULTS 29](#_Toc155621215)

[IMPORTANT SOFTWARES AND FRAMEWORK USED 29](#_Toc155621216)

[KERAS 29](#_Toc155621217)

[Flask 30](#_Toc155621218)

[Tools 30](#_Toc155621219)

[3.2 RESEARCH DESIGN 32](#_Toc155621220)

[3.2.1 TYPE OF RESEARCH 32](#_Toc155621221)

[3.2.2 VARIABLES 32](#_Toc155621222)

[3.2.3 DATA COLLECTION METHODS 32](#_Toc155621223)

[3.2.5 ANALYSES METHODS 33](#_Toc155621224)

[Chapter : 4 34](#_Toc155621225)

[4.0 RESULTS AND DISCUSSION 34](#_Toc155621226)

[Chapter: 5 35](#_Toc155621227)

[5: CONCLUSION 35](#_Toc155621228)

[REFERENCE 36](#_Toc155621229)

[APPENDICES 39](#_Toc155621230)

[ETHICAL CLEARANCE 39](#_Toc155621231)

[Figure 1, evolution of CNN Models 8](#_Toc155616083)

[Figure 2 LeNet n7 9](#_Toc155616084)

[Figure 3 dermatological diseases (n8) 9](#_Toc155616085)

[Figure 4 CNN with deep structure(n7) 10](#_Toc155616086)

[Figure 5 GoogLeNet (n7) 11](#_Toc155616087)

[Figure 6 train\_datagenerator 21](#_Toc155616088)

[Figure 7 multiplication and summation in a convolutional layer, source 15 22](file:///C:\Users\hipho\Desktop\desertation\Desertation%20Report%20com726,%20SOLENT%20UNIVERSITY.docx#_Toc155616089)

[Figure 8 model development work flow 24](#_Toc155616090)

[Figure 9 Model evaluation 25](#_Toc155616091)

[Figure 10 hyper parameters 25](#_Toc155616092)

[Figure 11 web App 28](#_Toc155616093)

[Figure 12 prediction result of a healthy skin image as healthy, original image) 28](#_Toc155616094)

[Figure 13 prediction result of a healthy skin image as affected(10% quality) 28](#_Toc155616095)

[Figure 14 Result checking and marking for analysation 29](#_Toc155616096)

[Figure 15 input images for the study 33](#_Toc155616097)

# LIST OF TABLES

Table 1 important Softwares and Tools………………………………………….31

Table 2 results………………………………………………………………………………..34

Table 3 results…………………………………………………………………………………34

# ACRONYMS

AI: Artificial Intelligence

API: Application Programming Interface

ANN: Artificial Neural Network

CNN: CONVOLUTIONAL NEURAL NETWORK

VGG-16: VISUAL GEOMETRY GROUP 16

ISIC : (International Skin Imaging Collaboration

MLP: MULTI-LAYER PERCEPTRON

NN: Neural Network

SVM: Support Vector Machine

CE: “Conformite Europeene”

ISO: International Organization for Standardisation

TGA: Therapeutic Goods Administration

# Chapter 1

## Introduction

This chapter outlines the background (section 1.1) and problem context (section 1.2) of the research, and its purposes (section 1.3). Section 1.4 sets the scope of the research work and provides objectives to achieve in this work, followed by section 1.5 which outlines the research questions considered to find answers for. The ethical considerations are discussed in section 1.6. Finally, section 1.7 includes an outline of the remaining chapters of this dissertation.

# 1.1 Background

To begin with, when it comes to the skin disease and its identification, there are more than one method available. One of the methods of detecting skin diseases is with the help of Artificial Intelligence. Although the concepts of deep learning were formulated in the 19th century, their applications have not yet become widely available to the public. Due to the technology and digitization boom, the amount of data collected is also significantly high and storing this digital data is now cheaper than before. Every individual probably has a huge collection of data on laptops, PCs and portable storage devices such as USB drives and external hard drives. Easy data storage, low-cost storage systems and computing devices are driving the use of artificial intelligence systems. The invention of advanced and high performance graphics processors can give an extra boost to this situation. The invention of advanced and high-performance graphics processors could add further development to this situation.

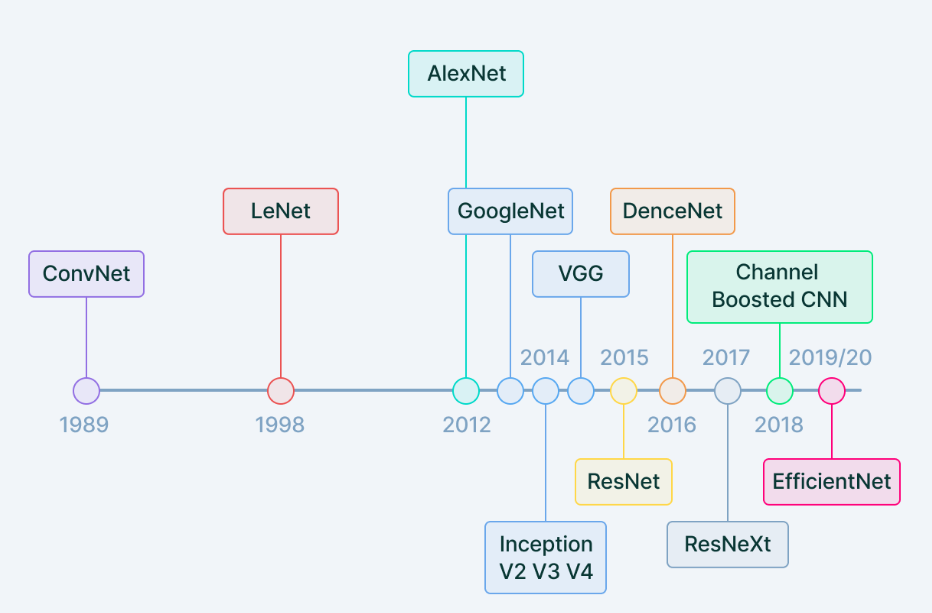


Figure 1, evolution of CNN Models

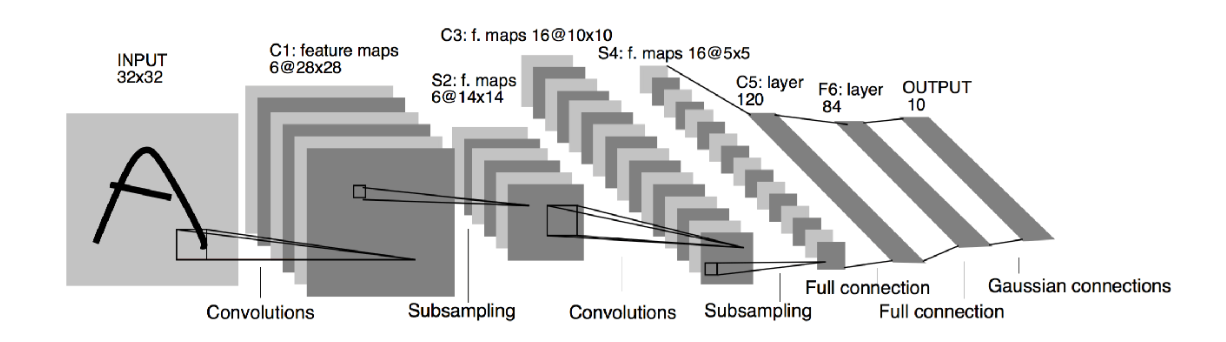


Figure 2 LeNet 25

Furthermore, the most common diseases in the world are dermatological diseases and the concerns arising from them play an inevitable role in human life. Although this condition is common, diagnosis is difficult and requires extensive expertise. In a year, 24% of the population consults a general practitioner for skin problems. Undergraduate dermatology curricula are inconsistent, suggesting that trainees should reassess their current knowledge and skills (13). Skin cancer is the most common type of cancer in the United States, with more than 9,500 people in the United States diagnosed with skin cancer every day. About 1 million Americans have melanoma (a type of skin cancer) (2). Skin cancer is a serious disease caused by changes in the properties of normal skin cells that become malignant and, due to DNA damage, the cells divide into abnormal forms. From the histopathological point of view of skin cancer, we can identify irregular structures with cell differentiation at different levels of chromatin, nucleus, and cytoplasm (4). The most dangerous skin cancer is melanoma. (5) Acne affects 50 million Americans each year, starting during puberty and affecting many young adults and adolescents. (2). Skin diseases not only cause damage to the skin but also lead to problems in daily activities (3), affect self-confidence and can lead to depression. This type of disease must be detected in the early stages to be analysed carefully because it can endanger human life.



Figure 3 dermatological diseases (26)

Nowadays, almost all sectors as well as a wide variety of fields are benefited from the use of computerized systems. Several tasks in the medical field can be made more convenient and efficient by using computer-aided tools. With the use of new technology, modern medical science is searching for solutions that could help doctors with any aspect of their work. Digital image processing and data mining are two popular methods in these fields. (2) The AI based image classification systems are used in medical field, object detection in satellite, traffic control and many more. For image classification task supervised machine learning is mostly used. An effective deep learning architecture which is frequently utilized in this domain is the Convolutional neural network and among this, the visual geometry group architecture called VGG16 is an advanced image classification model which is specially designed for image classification, which achieved top 5 test accuracy in ImageNet (a data set contains more than 14 million images that belongs to 1000 classes). (27) and which is known for its depth and performance in image classification tasks. As the demand for accurate image classification systems continues to grow in various fields, it is of paramount importance to understand the potential impact of input image quality on the performance of such models.

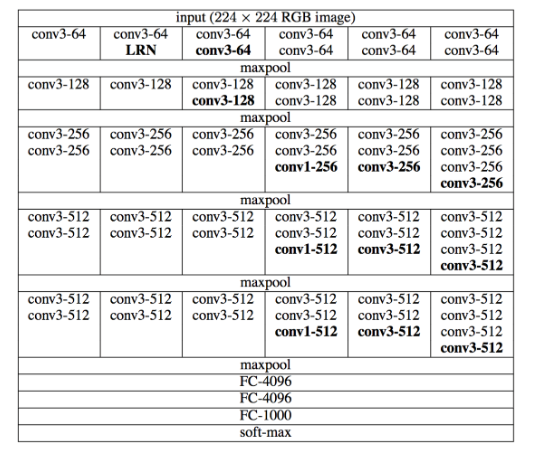


Figure 4 CNN with deep structure(25)

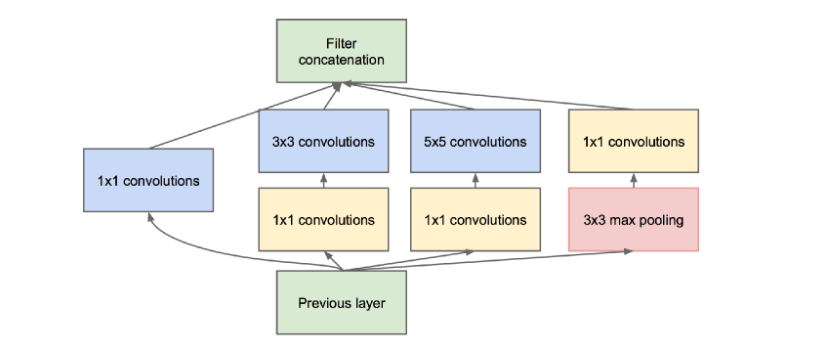


Figure 5 GoogLeNet (25)

# 1.2. Problem Context

This master's thesis addresses the exciting question "Is there a difference in the output of an image classification system based on the VGG-16 model when images of different quality are uploaded primarily for the same image? The study of this research question aims to reveal the specialized interactions between the VGG-16 model and image quality changes, ultimately giving information about the model's adaptability and robustness in real-world scenarios.

# 1.3 Purpose

The purpose of this of this research is to inform and improve image classification systems to ensure optimal performance across a wide range of image qualities. Answering this question has implications for many applications including medical imaging, surveillance, and autonomous systems where image quality can vary. In addition motivation behind this research comes from real-world challenges posed by images with varying qualities. In real-world scenarios, image quality is rarely uniform, and it is important to understand how well a model can cope with this diversity, in order to deploy a reliable image classification system.

# 1.4 Proposed solution

The proposed solution involves creating an image classification system deployed as a web application. In this web application, copies of images with different image quality will be uploaded to analyse the classification results. The goal is to deeply study how the VGG-16 model responds to changes in image quality and derive meaningful insights into its adaptability and robustness in real-world scenarios. Additionally, the results obtained from the web application are meticulously analysed to provide a comprehensive understanding of the model's performance under different image quality conditions. The lessons learned from this analysis provide valuable insights to the field of image classification systems, especially in situations where image quality is variable.

# 1.5 Aim, Objectives and Scopes

# Aim

The aim of this study is to develop a web app that can classify a skin disease from normal skin using deep learning technology to analyse and find the behaviour of the system on different quality images.

The scope of this thesis is to find the impact of impact of image quality difference in terms of memory and clarity on the performance of the trained VGG-16 model. The main focus is on evaluating how the changes in image quality together with different levels of resolution and compression, influence the prediction results of the of an image classification system. This study explores the practical implications of these variations, identifying that the image quality adjustment in real world cases may directly impact the storage requirements. The adjustments of image resolution is performed using python imaging libraries such as open pillow, open cv. Additionally the resizing options in the windows photos app is used to create images of different quality and memory. The objectives of this study is listed below.

* Collect images of the disease affected skin.
* Pre-process the images.
* Create a deep learning model for classifying the images.
* Train the model with the collected images.
* Develop a web based skin disease classification system.
* Conduct an analysis of results, by uploading the same image with different qualities.
* Check there is any difference in results when uploading images of different quality.

Research question: Is there any difference in the output of the image classification system based on a VGG-16 model when images of different qualities (different memory) are uploaded mainly for the same image.

# 1.6 ETHICAL CONSIDERATION

This research focuses on the analysis of image data from the ISIC dataset, conducted with a commitment to ethical standards. Although the research does not involve direct interaction with human participants, several ethical considerations are paramount: The ISIC dataset is a valuable resource provided by researchers and dermatologists around the world. In using this dataset, the study commits to giving due credit and recognizing the efforts of those who contributed to its creation. The images downloaded from ISIC data set are without any personal details and only contains a unique code. Any specific license or terms of use related to the ISIC dataset will be strictly adhered to. The study will comply with the conditions of use specified by the ISIC data set. This includes compliance with any restrictions on use, redistribution or modification of the dataset. All image data will be handled responsibly, with an emphasis on preventing unwanted consequences or misuse. Appropriate measures will be taken to secure the dataset and ensure that it is used only for the intended purposes of the study. In addition the web app and its underlying components were designed without any bias

# 1.7 DESERTATION OUTLINE

In the following sections the existing literature related to this study is discussed and the gap in literature is also mentioned. Followed by this, in chapter 3, the methodology and research design is clearly explained. In chapter 4 the results are discussed and the chapter 5 ends with the conclusion of this study

# Chapter 2

## LITERATURE REVIEW

# 2.1 Topic 1: Automated Systems for Skin Disease Detection

# 2.1.1 Subtopic 1: Mobile Applications for Skin Disease Diagnosis

In (9) SkinVision app is a medical device controlled by AI and expert skin health experts. It has been developed in collaboration with dermatologists for clinical use as part of skin vision service founded in 2011 and its chief Executive officer is Eric De Heus. An Android software called "SkinVision" was suggested for the diagnosis of melanoma skin conditions. The skinVision app sensitivity rating was 95 % that means this algorithm can accurately detect skin cancer 95% of the time.(19) The most attractive point about skinvision is, it expands a person’s ability to self-examine their own skin and provide the knowledge of when, how, and why to act.

The SkinVision algorithm's machine learning process is powered by data consisting of over 100,000 skin spot images selected from 2.9 million user photos that were previously evaluated by a team of dermatologists and categorized correctly according to the corresponding diagnosis. This ensures a large, diverse and well-maintained database covering all skin types and conditions. From there, SkinVision data scientists trained an algorithm on these images, allowing it to detect 4,444 patterns between photos of skin blemishes and risk labels created by dermatologists. The result is a powerful tool that uses these patterns to predict the risk level of future skin blemish photos provided by users. To ensure the quality of the risk assessment, the algorithm is tested against the gold standard of skin cancer: photographs of skin spots identified as skin cancer by biopsy (this is a procedure that identifies all types of skin cancer by examining skin tissue under a microscope)

The app allows users to snap a picture of the disease spot using their phone's camera and upload it. It then determines the risk level in less than 30 seconds as low, medium, or high. SkinVision is a clinically validated and regulated medical product. It has CE mark, TGA approval and ISO certification, among others. This app is recommended by doctors, pharmaceutical companies, and health insurance companies around the world. It is also recommended by melanoma patient support and advocacy groups as an important part of a self-examination tool for skin cancer prevention.

# 2.1.2 Deep Learning Models for Skin Disease Detection

A support vector machine was used in the proposed melanoma skin cancer detection model in [12]. Poornima M.S.and Dr. Shailaja K. have made significant contributions to melanoma detection using image processing tools, active contour segmentation, local binary pattern (LBP) feature extraction, and support vector machine (SVM) classification. Their methodology focuses on early detection through the extraction of features such as colour attributes, lesion shape, and texture. SVM-based classification systems stand out as powerful tools to differentiate between non-melanoma and melanoma lesions. This study is one of the the basis of an investigation into the influence of input image quality on melanoma prediction.

The literature highlights the high incidence of skin cancer, especially melanoma, in India and the need for cost-effective automated early detection systems. Existing studies highlight the urgency of introducing efficient diagnostic tools due to the increasing prevalence compared to other countries. This work extends this perspective by considering quality variations in the input images and understanding how they affect the prediction results. Feature extraction techniques such as colour attributes and texture analysis, proposed by Shailaja K., play an important role in conventional melanoma detection. These features serve as important features to distinguish benign and malignant lesions. While the work in 12 emphasizes the importance of manually developed features, this work departs from this by examining the adaptability of deep learning models to different input image qualities.

Similar to this, the use of convolutional neural networks for the detection of five various skin diseases is included in the work of Jainesh Rathod, Vishal Waghmode, Aniruddh Sodha, Drs Bavatankar of the Sardar Patel Institute of Technology in Mumbai Their paper proposes an automatic image-based system for skin disease detection using machine learning classification, specifically convolutional neural networks (CNN). The accuracy of the suggested system was 70%. However, the authors came to the conclusion that accuracy may be raised above 90% by utilizing a higher dimensional dataset [13]. The purpose of this system is to overcome the challenges associated with the complexity of dermatological diagnosis and the diversity of physician expertise. The system addresses the complexity of dermatological diagnosis by automating image analysis, noise removal, and enhancement processing, ultimately leading to efficient disease classification.

The proposed architecture includes important phases such as image acquisition, preprocessing, data storage for training and testing, and a classifier to identify skin disease types. The CNN-based feature extraction process is highlighted as a key component, highlighting the ability to analyse and process image data based on various features. This includes removing unwanted noise, enhancing the image, and classifying the image using a softmax classifier.

However, while these studies provide promising opportunities for automated diagnosis of skin diseases, a literature review reveals a gap in research regarding the impact of input image quality on the performance of such systems. It is worth noting that this forms the basis of his current research to investigate whether variations in image quality affect the prediction results, especially for his web apps that use his VGG16 architecture. The robustness of machine learning algorithms, especially those using feature extraction, can be affected by the quality of the input images, highlighting the need to understand these ideas.

# 2.1.3 Symptom based classification of Skin Diseases

This (16), article describes the application of data mining algorithms to classify skin diseases based on input symptoms which is done by M. Sudha and B. Poorva. About 366 multivariate instances and 34 attributes were used (data from the UCI repository, for example, the family history feature has the value 1 if any of these diseases have been in the family, and if not, it is zero). For the classification of six diseases, they used Naïve Bayes, Random Forest, SVM Linear Regression, and the KNN algorithm, which gives 99.31%, 97.80%, 94.35%, 82.14%, and 94.44% accuracy, respectively. From this paper, in the case of data with instances and attributes, we can conclude , the most suitable tool for dermatological issues, is the probabilistic classifier, and linear regression can be used to check whether a disease is present or not. This algorithm will help the doctors classify the disease correctly when it can give a good accuracy.

The literature supports the application of Naïve Bayes in medical data mining, highlighting its efficacy when attributes are considered independent. The integration of data mining approaches and image processing techniques for predicting skin diseases is a notable aspect considered in this study. This integration allows the extraction of meaningful features from digital images of the skin, contributing to more accurate disease classification.

# 2.2 WEB-BASED SYSTEMS FOR SKIN DISEASE DETECTION

# 2.2.1 ADDRESSING HEALTHCARE GAPS THROUGH WEB-BASED SYSTEMS

An article titled “Web-based skin disease diagnosis using convolutional neural networks” by Samuel Akyeramfo-Sam et al. published in the International Journal of Information Technology and Computer Science in November 2019(17).It deals with the important issue of human skin diseases. Especially in sub-Saharan Africa. Highlighting the tedious and time-consuming nature of traditional methods for diagnosing skin diseases, the authors utilized convolutional neural networks (CNNs) to diagnose conditions such as atopic dermatitis, acne vulgaris, scabies and dermatitis. They propose a web-based system called medilab-plus'' for efficient and rapid detection. Furthermore, this study focuses on the Ghanaian context where dermatological processes are not yet automated. Due to this the diagnosis is not accurate and is very time consuming. The data for the training and testing of the CNN model was collected from Sunyani Municipality, Ghana, and the pre-processing, image segmentation, and feature extraction were carried out on this data set to get the results. According to this paper, this web-based system can provide a more realistic and reliable service for skin disease detection, reduce the diagnosis time, and provide fast and accurate health care. The proposed CNN-based system showed a classification accuracy of 88% for atopic dermatitis, 85% for acne vulgaris, and 84.7% for scabies. The computation time of this system is very short (0.0001 s), suggesting that it has the potential to significantly improve the efficiency of dermatological diagnosis compared to manual methods.

The literature review section of this paper provides an overview of common skin diseases in Ghana, discusses the application of machine learning in the diagnosis of skin diseases, and reviews related studies. The authors highlight the importance of automated systems for timely intervention due to the prevalence of skin diseases and limited number of dermatologists in Ghana. This study contributes to the existing literature by proposing a web-based solution tailored to the Ghanaian context and demonstrates the potential of CNNs in dermatological diagnosis.

In summary, this paper highlights the importance of leveraging technology, particularly CNNs, to address the challenges associated with the diagnosis of skin diseases in Ghana. The proposed system medilabplus represent a valuable contribution to the field of dermatology, as it not only shows promising accuracy but also provides a rapid and efficient alternative to traditional diagnostic methods. However this study did not have any particular focus on the variation of result based on image quality difference.

# 2.2.2 RULE-BASED SYSTEMS FOR SKIN DISEASE RECOGNITION

In paper 18, the author describes a system that allows users to recognize children's skin diseases via an online system and gives them useful suggestions. It is a rule based system with an if/then structure, and it can identify about 7 types of skin disease in children. In the user friendly interphase, there are five types of modules: the login module, the module of diagnosis, the info module, the management module, and the report module. Some of these modules are used for identifying diseases based on certain questions and managing the system. After the user completes the questions in the diagnosis module, they can see the skin disease that affected the children, the symptom as per the questions answered by the user, and its possible treatment.

# 2.2.3 Diagnosing skin diseases on mobile devices: Integrating image processing, data mining and user interaction

Research paper mentioned in 2, addresses the important issue of skin diseases prevalent in Sri Lanka due to climatic and living conditions, it highlights the importance of early diagnosis and treatment of skin diseases to prevent complications and emphasizes the need for computer assisted tools in the field of medical science. The authors highlight the impact of skin diseases on the physical and psychological health of individuals and they propose an expert system for the diagnosis of skin diseases using image processing and data mining techniques. The purpose of this system is to quickly and accurately detect skin diseases and provide users with timely advice and treatment. The process involves users uploading images of their affected skin to the system and answering questions about their skin conditions and symptoms. The system uses image processing techniques such as adaptive histogram equalization (contrast improvement), Gaussian operators, morphological transformations, and watershed algorithms to achieve noise reduction and image enhancement. Images are post-processed to improve their shape and edges. In the data mining unit, features extracted from the image processing stage are analysed to identify skin diseases. The system uses five different data mining classification algorithms (AdaBoost, BayesNet, J48, MLP and NaiveBayes) to predict skin diseases, such as eczema, impetigo, and melanoma.

The article discusses related work in computerized skin disease diagnosis systems, highlights the limitations of existing solutions, and presents their novel approach.

It compares different systems based on reliability, speed, user-friendliness, scale, details of results, and methods used. Limitations of the proposed system include a focus on three specific skin diseases, development only for Windows applications, and a fixed distance between the camera lens and the affected skin area during imaging.

The classifier's accuracy for 3, diseases was calculated as the percentage of correct results compared to the total number of test sets. Depending on the accuracy of the data, MLP and J48 have more reliable predictions. Further analysis of the comparison between J48 and MLP they found for eczema, the best classifier is J48 and not MLP for certain training datasets. However, when looking at the overall correctness ranking, the author found that the maximum overall correct result is given by MLP. Therefore, to get the most optimal results for the entire system, they decided to use the MLP classifier. In this expert system, 85% of eczema cases are correctly identified, 95% of impetigo cases are correctly identified, and 85% of melanoma cases are correctly identified.

In summary, the research paper presents a promising expert system for diagnosing skin diseases, integrating image processing and data mining techniques. The authors provide an in-depth assessment of the system's accuracy and suggest future improvements, such as cross-platform compatibility and language support. The article contributes to supporting the field of medical informatics, providing potential solutions to improve the ability to diagnose and treat skin diseases early. Although there are some drawbacks, the authors’ note that their system is highly reliable and performs well, as it incorporates both image based and questionnaire related methods to improve its accuracy.

# 2.2.4 TRANSFER LEARNING WITH VGG-16 for SKIN DISEASE CLASSIFICATION

Transfer learning has become a powerful technique in machine learning, especially in the field of image classification. Traditionally, machine learning algorithms are designed to operate independently on specific feature spaces and distributions. However, in real situations, the assumption of identical feature space between training and testing data is often invalid, requiring models to be rebuilt from scratch in case of changes. Change and distribute characteristics. Srikanth Tammina, in the paper titled “Transfer Learning Using VGG-16 with Deep Convolutional Neural Networks for Image Classification” (2019), addresses this challenge by introducing transfer learning as a method to reuse knowledge from pre-trained model for different tasks. The article focuses on image classification and uses the VGG-16 model with a deep convolutional neural network (CNN) for this purpose.

The article begins by highlighting the inherent ability of humans to transfer knowledge between different activities and draws parallels with machine learning. Traditional deep learning models face challenges data for training and testing is less for the target domain. Transfer learning becomes important in such cases, using pre-trained models and knowledge from related domains to improve performance in the domain of interest. The author presents a convolutional neural network (CNN) model designed using Python and TensorFlow. The choice of the VGG-16 model was motivated by its architecture, which includes 13 convolutional layers and 2 fully connected layers. A model trained for image classification using knowledge extracted from pre-trained models. The paper addresses an image classification problem involving a small number of category learning samples, focusing specifically on the classification of dog and cat images. The goal is to build an effective model under the constraints of limited training data. The CNN architecture consists of convolutional layers, nonlinear activation functions (ReLU), pooling layers, and fully connected layers.

The article highlights the use of transfer learning with pre-trained models like VGG-16, trained on large datasets like ImageNet. The pre-trained model is used as an effective feature extractor, allowing to transfer knowledge about low-level features such as space, edges, rotation and illumination to the target problem. The dataset includes 25,000 images of dogs and cats, with a subset of 5,000 images used for training and 2,000 images for validation. The scarcity of labeled data in the target domain motivates the use of transfer learning to improve classification accuracy. The author performs a comprehensive analysis, comparing the performance of the base CNN, image augmented CNN, and fine-tuned CNN with the pre-trained VGG-16 model. The results show a significant improvement in accuracy, with the transfer learning method achieving the highest validation accuracy of 95.40%. The paper proposes potential future improvements, such as exploring other pre-trained models, extending the approach to different domains, and studying the impact of transfer learning on with different image classification tasks.

In summary, Srikanth Tammina's paper highlights the effectiveness of transfer learning using VGG-16 for image classification, demonstrating its potential in solving data-related challenges. Labelled data is restricted to specific domains. This approach paves the way for further research on leveraging pre-trained models to improve the performance of machine learning models in various applications.

The article titled "Very Deep Convolutional Networks for Large-Scale Image Recognition" by Karen Simonyan and Andrew Zisserman, presented at ICLR 2015, investigates the impact of convolutional network depth on the accuracy of large-scale image recognition. The authors focus on evaluating networks of increasing depth, especially using small (3x3) convolutional filters. The most important contribution of this study is that we demonstrate that significant improvements in accuracy can be achieved by increasing the depth of the weight layer from 16 to 19. The authors describe the architecture of ConvNet in detail, highlighting the use of small receptive fields, a tuning layer with ReLU activation, and the absence of local response normalization (LRN) for improved performance on the ILSVRC dataset. Configurations range from 11 to 19 weight layers, with a different number of channels for each convolutional layer. The authors show that increasing the network depth improves accuracy, especially for small convolutional filters. This insight into network architecture can serve as a basis for understanding how model depth affects the predictive results of image classification systems.

Although this study deals with deep study of the significance of the network depth, it does not specify the behaviour of system under different image qualities specifically.

# Summary and Implications

The literature review provided insights into various automated skin disease detection systems, including mobile applications, deep learning models, symptom-based classification, and web-based solutions. Although existing research is promising, there is a clear gap in research regarding the impact of image quality on the performance of these systems. This study aims to fill this gap by studying how variations in image quality affect the results of automated systems for diagnosing skin diseases.

The implications of this study are important for advancing the field of automated skin disease detection. Understanding how image quality affects the performance of these systems will help improve the robustness and reliability of such tools. The results will guide the development of more effective algorithms and help optimize existing systems for different image qualities. Furthermore, this research also contributes to the broader goal of improving the accessibility and accuracy of skin disease diagnosis through automated technologies.

# 

# Chapter 3

# 3.1 METHODOLOGY

## 3.1.1 Collecting the data

For this study, the data that is the images of healthy and disease affected skin is collected is from ISIC (International Skin Imaging Collaboration) dataset. ISIC dataset contains images of different types of skin lesions which are contributed by researchers and dermatologists from different parts of the world. It is a publicly assessable data that can be assessed from an online website. ISIC's main goal is to support efforts to reduce melanoma-related deaths as well as unnecessary biopsies by improving the accuracy and efficiency of early melanoma detection. The objective of ISIC's work is to develop a digital imaging standard that integrates dermatology and informatics, with the aim of improving diagnostic accuracy through AI. Although ISIC's initial focus is on melanoma, the goals it pursues are important for advancing the broader landscape of skin imaging and artificial intelligence in dermatology. These include non melanoma skin cancers and inflammatory skin diseases. (28)

The input data size for the VGG-16 image classification model is a 224x224 image, it is better to crop images into these sizes for better results. (14). The images for training, testing and evaluating can be found from many sources, such as HAM10000, ISIC, PH2 Dataset, Dermofit Image Library etc.

## 3.1.2 Pre-processing of collected images.

To ensure that our model never sees the exact same picture twice it is desirable to augment the training set of samples by applying several random changes. The model can generalises better and overfitting is reduced as a result. The Image Data Generator class in Keras allows for this to be accomplished. When training, rescaling the pixels to normalize their values within the range [0, 1] is essential for optimal convergence. Additionally, an 80/20 validation split is implemented, with 80% of the data being provided for training and 20% for validation.

Data augmentation, an optional step, introduces variation to the training data set and improves the model's generalization ability. Augmentation techniques include random rotation, horizontal flip, width and height shifting, shear transformation, and random zoom. The dataset is further split into training and validation sets based on the specified validation split. During model training, pre-processed training data is fed to the VGG-16 model. When enabling the data augmentation, training data set undergoes additional transformations to increase its diversity. The training process is conducted for a fixed number of epochs, with validation steps occurring at regular intervals. Optimal training efficiency is achieved through the use of callbacks, including model check pointing and early stopping.

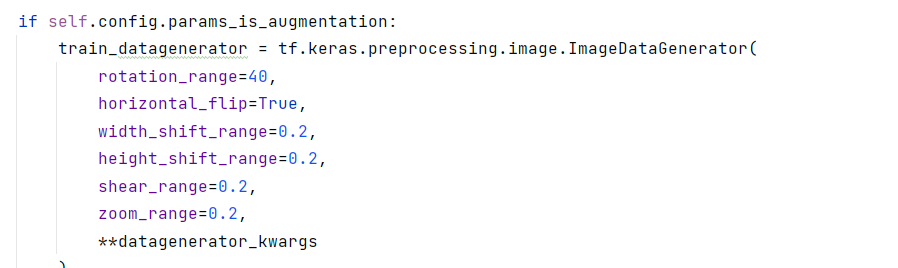


Figure 6 train\_datagenerator

The training data generator applies the configured data augmentation techniques, and the validation data generator prepares a validation data set for evaluating model performance. After training, the model is evaluated on another set of data that it has never seen before. Evaluation includes loading the trained model, setting up a validation data generator, and evaluating the model's performance against relevant metrics. The VGG-16 model has its own pre-processing and does not require much pre-processing because of the deep structure of the architecture. The simplicity because of the less significance of pre-processing methods is a peculiarity of this model and it is because of the deep structure of the model.

# Image data generator

# 3.1.3.1 MODEL ARCHITECTURE

### VGG-16 ARCHITECTURE

VGG-16 is a CNN model that includes 13 convolutional layers and 3 fully connected layers and other components are activation functions and pooling layers.

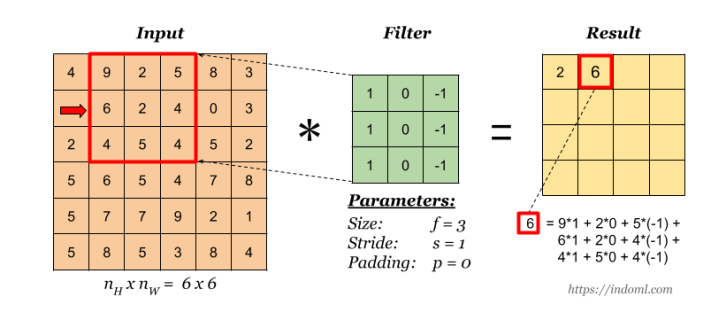
1. Convolutional layer: In a convolutional layer of a VGG16 model, to create a feature map for the next layer, a kernel matrix is passed over the input matrix. A mathematical operation called convolution is done by sliding the kernel matrix over the input matrix. This particular type of linear operation called convolution is frequently used in many different fields, including image processing. (15) The disadvantage of the feature map output of the convolutional layer is that it records the exact locations of the objects in the input. This means that while cropping, rotating or any other small change to the input image will result in a completely different feature map. To solve this problem, we address the down sampling of convolutional layers. Down sampling can be achieved by applying a pooling layer after the nonlinear layer.

Figure 7 multiplication and summation in a convolutional layer, source 15

B) Pooling layer:

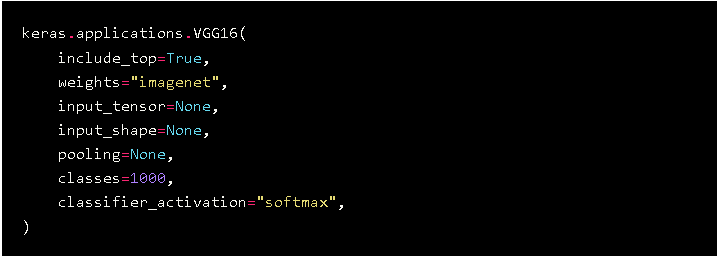
Pooling helps make the representation nearly invariant to small translations of the input. Translation invariance means that if we shift input a little, the values ​​of most of the grouped outputs do not change. The majority of the pooled output’s values remain constant even when we slightly alter the input. There are two types of pooling max pooling; and average pooling. Pooling layers play an important role in sub sampling the spatial dimensions of the input data, facilitating the extraction of important features while reducing computational complexity. In the VGG-16 model, max pooling is applied after the sets of convolutional layers. Max grouping involves selecting the maximum value from a set of values ​​in a specified window and dragging the window over the input. This preserves the most prominent features of local regions, provides a form of spatial invariance, and helps the network focus on the most meaningful information. (15)

# 3.1.2 MODEL DEVELOPMENT

## 3.1.2.1 VGG16 MODEL

Visual Geometry Group is a deep convolutional neural network architecture with many layers, in VGG-16 there are 16 convolutional layers. This model is the base of the object recognition and image classification models and is still playing an important role in most popular image recognition architectures. This CNN model was proposed by A. Zisserman and K. Simonyan who are from the University of Oxford. The VGG16 model is one of the popular models submitted to ILSVRC-2014, and it is able to get an accuracy of 92.7% among the top 5 test accuracy in the ImageNet data dataset, consisting of more than 14 million images. The classification of 1000 object categories is made possible by the powerful 16-layered VGGNet-16, including animals, mouse, skin disease images, etc. (14). This model can be used flexibly.

# 3.1.2.2 VGG16 Function



The parameters named include top, weights, input tensor, input shape, pooling, classes and the classifier activation are the parameters that allows to customize how the vgg16 model is initiated.

* Include top: The inclusion of the fully-connected layers at the top of the network is indicated by this boolean value. The model will incorporate the original VGG16 classification layers if set to True. You can add your own custom classification layers if this option is set to False. For the study this option is set to false and one custom classification layer is created
* Weights: The source of the initial weights is specified by this option. It can accept variables such as the location to a particular weights file, none (random initialization), or ImageNet (pre-training on ImageNet). In this study the weights are set as ImageNet weights
* Input shape: The input images' shape is specified by this option. In the event that include top is False, it must be given. (224, 224, 3) is the default value if it is not supplied. For this study the input is specified as (224, 224, 3).
* Classes: When include top is True, this argument is used. It indicates how many classes are involved in the classification task. Since ImageNet has 1000 classes and VGG16 was first trained on it, 1000 is the default. For binary classification it can be set to 2, because this study is based on binary classification value of number of classes is set to two.
* Input tensor: input tensor can be specify as a Keras tensor as the model's input by using this optional parameter. The model will use input shape to generate an input tensor if it is not given and this is the case for the model used for this study
* Pooling: When include top is False, an optional pooling mode is used for feature extraction. Three settings are available: max, average, and none. None indicates that the final convolutional block's 4D tensor output will be the model's output. ‘Avg’ indicates that the last convolutional block's output will undergo global average pooling, resulting in a 2D tensor as the model's output. Max denotes the application of global max pooling. The model for this study uses max pooling.
* Classifier activation: The activation function that is used on the model's "top" layer is indicated by this parameter. The layers that make the final predictions are usually dense (completely linked) and make up the "top" layer. Activation functions are essential to the model's capacity to recognise intricate patterns in the input since they impart non-linearity to the network. The classifier activation parameter is only relevant if the top layer (fully connected layer) is involved. If include top is set to False, it means that you are using the model for feature extraction and the final classification level is not included. If classifier activation is set to none, the model will output the raw logits from the top layer. The logit is the unnormalized prediction before applying the activation function.

# 3.1.2.3Model Development Work Flow

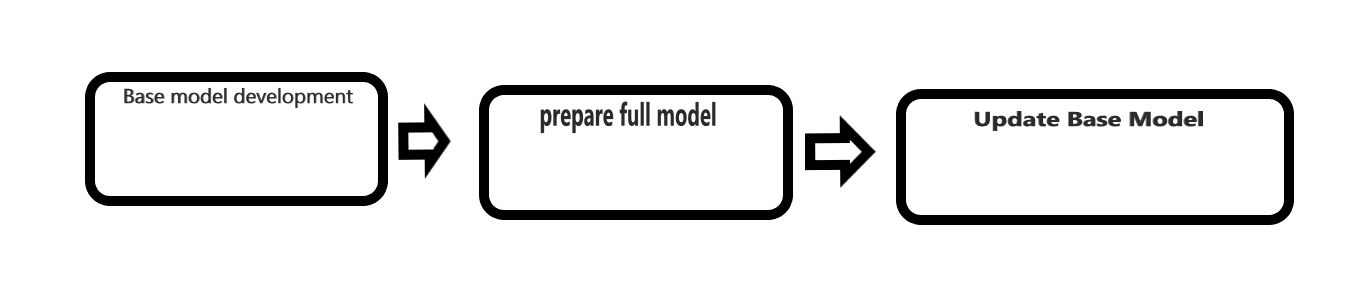


Figure 8 model development work flow

This class is initialized with a configuration object (“Prepare BaseModel Config”) to set various parameters for model preparation. Prepare BaseModel allows to easily prepare and customize base models using the VGG16 architecture of tensor Flow’s Keras API. The get base model method initializes a VGG16 model with the specified input shape, pretrained weights, and upper layer embeddings and the resulting model will be saved to the specified path.

The \_”prepare full model” method is a static method that customizes the loaded VGG16 model. In this method the output from the base vgg16 model is smoothed using sigmoid activation function, which is a suitable activation function for binary class classification and added a dense layer with the specified number of classes. Additionally, all of the layers of base vgg16 model are frozen based on specified parameters (“freeze all” or “freeze till”)

A custom full model is compiled using a stochastic gradient descent (SGD) optimizer, a binary cross-entropy loss function, and various metrics such as accuracy, precision, recall, AUC, and binary precision. A compiled summary of the complete model is printed and the model is saved in the specified path. The “update\_base\_model” method summarizes the above steps, builds and compiles the complete model, and saves it.

## 3.1.3 MODEL EVALUATION

The main matrices used for model evaluation are accuracy and loss. For a good model, accuracy would be high and loss will be less. As per the hyper parameters these two matrices vary. The hyper parameter with the best values of these matrices selected for this project. Matrices such as accuracy, precision, recall, area under the ROC curve and Binary Accuracy were analysed during training. In evaluation stage loss and accuracy is calculated on the data set.

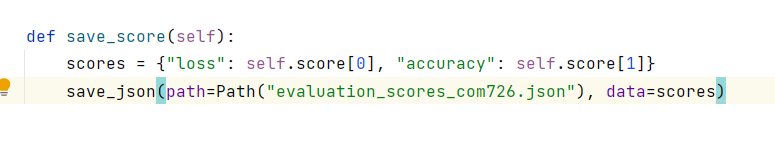


Figure 9 Model evaluation

## 3.1.4 Hyper parameter Tuning

Careful consideration and tuning of hyperparameters plays a key role in efforts to optimize the performance of deep learning models. The hyper parameters selected for this study are the result of a systematic investigation to achieve the best possible model.

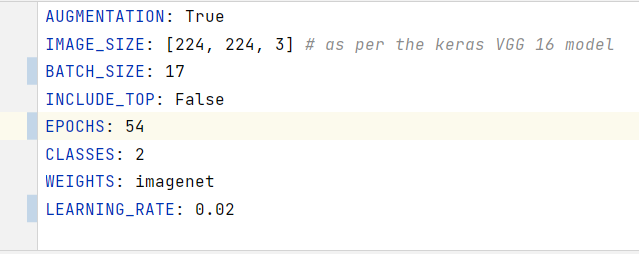


Figure 10 hyper parameters

Data augmentation

Data augmentation set to True was used to increase the diversity of the training data set. Various transformations such as rotation, scaling, and mirroring are applied to artificially increase the amount of training samples.

This makes the model more robust and generalizes better to unknown data.

Image size

The image size selection set to [224, 224, 3] is based on the Keras VGG16 model specification. This 224 x 224 pixel spatial size with three colour channels (RGB) is important for compatibility with pre-trained weights and architectures.

Batch Size

After trying with different batch sizes, batch size of 17 was set for the training process. This represents the number of training samples processed in each iteration. The selected batch size ensures a balance between computational efficiency and model convergence.

Top Layer Inclusion

The decision to exclude the top layer (fully connected layer) of a pre-trained model, as indicated by "INCLUDE\_TOP: False", depends on the specific task available. This makes it easy to adapt to the unique characteristics of your dataset.

Number of epochs

Training runs over 54 epoch. An epoch refers to a complete run through the entire training data set. The selection of 54 epochs is based on empirical observations of convergence behaviour and validation performance.

Number of Classes

In the current binary classification task, the model is configured to distinguish between two classes. This hyper parameter labelled CLASSES: 2 since it is a binary classification problem.

Pre-trained weights

The model is initialized using weights pre-trained on the ImageNet dataset, specified by WEIGHTS: ImageNet. By using pre-trained weights, the model can capture hierarchical features and accelerate convergence.

Learning Rate

The learning rate, set to 0.02, determines the step size during optimization.

This hyper parameter is important to control how quickly the model adapts to the training data. The chosen values ​​are the result of careful tuning to achieve a balance between convergence speed and stability.

In summary, the hyper parameters were carefully chosen to not only leverage the power of transfer learning with pre-trained weights, but also to create a model that is tuned to the complexity of the specific classification task.

3.1.7 CREATING INPUT DATA

In this stage, for finding, whether there is any difference in the output prediction or there is no change in the result from the web app, an accurately classified image is selected. Images of different qualities are generated from this accurately classified image by utilizing 5the default image editor in windows. The image editor named photos has an option to adjust the quality of the image



Using this feature, quality of an image can be varied in terms of percentage of quality that will change the quality from current quality of the image, as a result the memory alteration occurs as well. After adjusting the quality percentage of the selected image, different copies are saved in the suitable folder which can be used for the study.

## 3.1.3 Classification of images using web app.

The study is conducted with the help of a web app that can be opened in the web browser of local machine. The images of the skin which is prepared for the study would be uploaded to the web app and the web app can classify them as affected skin or healthy skin based on its learned parameters.

For this study, images of the same type of skin with different qualities are going to be uploaded and the results of the prediction will be marked and examined.

When creating the web app using the flask library, the image classification model and the necessary files and documents should be ready. In this study the web app is based on a VGG-16 model from the keras library, which is a pre trained model using the imageNet source. ImageNet image source is a large database of images with their labels. After adding some custom layers to the base model, the updated model can be saved. Because the pre trained model classifier has greater than two classes, and our task is to classify the images into two classes.



Figure 11 web App

# 3.1.7 UPLOADING TO THE WEB APP

In this step the images of diverse qualities are uploaded to the skin disease classification system’s web app.

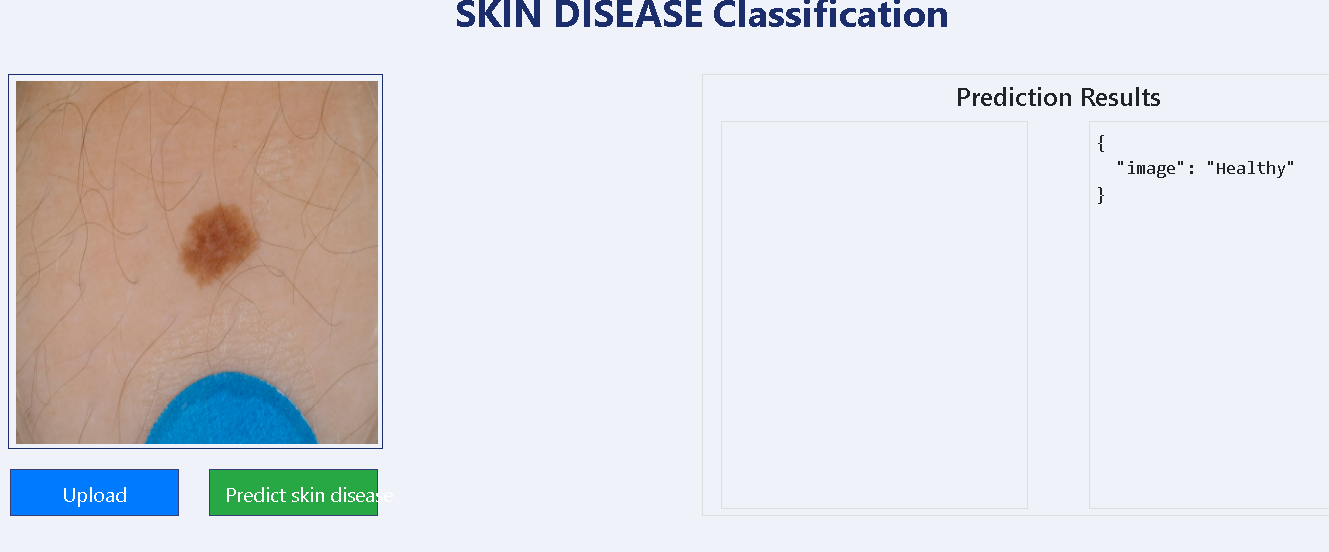


Figure 12 prediction result of a healthy skin image as healthy, original image)

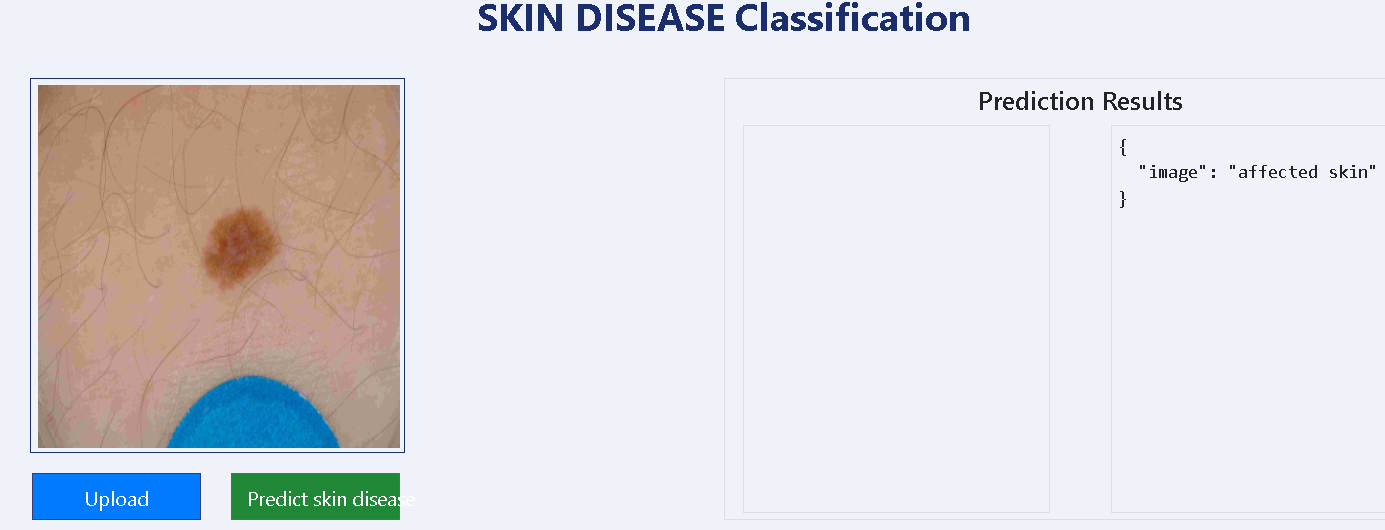


Figure 13 prediction result of a healthy skin image as affected(10% quality)

# 3.1.8 CHECKING AND MARKING RESULTS

Followed by the image upload, the result of each upload is accurately marked for analysing the results.

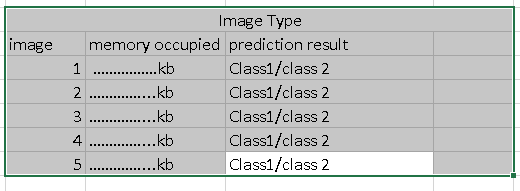


Figure 14 Result checking and marking for analysation

# IMPORTANT SOFTWARES AND FRAMEWORK USED

TENSORFLOW

TensorFlow, released by Google in November 2015, is a cornerstone in the field of open source frameworks for digital computing, large-scale machine learning, and deep learning. Running seamlessly on CPUs, GPUs, and Google's custom tensor processing units (TPUs), TensorFlow provides a flexible environment for developers, data scientists, and engineers. TensorFlow supports a variety of applications, from image recognition and natural language processing to handwritten digit classification and word embedding’s. While Python serves as a front-end API for application development, TensorFlow provides wrappers in many languages, including C++ and Java, ensuring compatibility across a variety of platforms. (20)

TensorFlow simplifies machine learning engagement with a high-level Keras API. Ideal for beginners, it makes it easy to quickly create and train models, providing an accessible starting point for those venturing into TensorFlow. For users requiring advanced control, TensorFlow supports the Keras functional API. This feature allows developers to create complex model topologies without compromising speed or performance. TensorFlow also supports an ecosystem of additional libraries and powerful models for testing, including Ragged Tensors, TensorFlow Probability, Tensor2Tensor, and BERT. (19)

# KERAS

Keras serves as a high-level API for the TensorFlow platform, providing an accessible and highly efficient interface for solving machine learning problems. Developed with a focus on modern deep learning, Keras covers the entire machine learning pipeline, including data processing, hyper parameter tuning, and deployment. Keras integrates seamlessly with TensorFlow, allowing users to take advantage of the scalability and cross-platform capabilities of the underlying framework. It can run on specialized hardware such as TPU Pods, large GPU clusters, and even export models for browser or mobile deployment. Keras's design philosophy is to enable rapid experimentation. This means users can start with simple, basic features, and as they gain more experience or need more advanced features, they can gradually drill down into more complex aspects of the library. This progressive approach accommodates users of different skill levels and promotes a smooth learning process. It’s simple and consistent interface minimizes the number of actions required for common use cases, provides clear error messages, and adheres to the principle of progressive complexity disclosure.

Keras's basic data structures are layers and models. A layer is a simple input/output transformation and a model is a directed acyclic graph (DAG) of layers. The tf.keras.layers. Layer class is the basic Keras abstraction layer. A Layer consists of a state (weight) and some calculations (defined in the tf.keras.layers call method). For pretrained model be called using tf.keras.applications.vgg16

# Flask

Flask is a web application framework written in Python and developed by Armin Ronacher and a team of international Python enthusiasts called Poocco. It is based on the WSGI Toolkit tools and the Jinja2 template engine, both projects under his Poocco umbrella. Web Server Gateway Interface (WSGI) is a widely used standard in Python web application development. It defines a common interface between web servers and web applications and serves as a specification to ensure interoperability within the Python ecosystem. Jinja2 is a widely used template engine for Python. In web development, template engines associate templates with specific data sources to dynamically render web pages. Flask is commonly referred to as micro framework, emphasizing its design philosophy of keeping the core of your application simple and scalable.

Flask is essentially a Python module that simplifies web application development with a small and easily extensible core. It features a micro-framework that intentionally avoids incorporating features such as object-relational managers (ORMs), making it a lightweight and flexible choice for developers. In contrast to Django, Flask is heavily based on Python. Flask boasts a low learning curve, making it easy for beginners to get started. Its high level of explicitness enhances code readability. With just a few lines of code saved in a resulting .py file, a simple web application can be created. Upon execution, a web server, initially accessible only locally, is launched. The application can then be accessed at 'localhost' on port 5000 through a web browser.

# Tools

Image data generator

In order to produce an output that exclusively contains newly transformed data, Keras ImageDataGenerator is used to take the original data as input, transform it randomly, and then output the resultant. The data is not added by it. In order to get an overall increase in the model's generalisation, data augmentation is also performed using the Keras picture data generator class. In the field of real-time data augmentation, the Keras image data generator is used to generate batches including tensor image data. When we use Keras' image data generator, we can loop through the data in batches. The image data generator class has a number of methods and arguments that are useful in defining how the data is generated. Using an image data generator, random operations like rotations, translations, shearing, scale adjustments, and horizontal flips are performed during data augmentation. (n5)

Feature-wise centre: This Boolean value is utilised to feature-wise set the input value for a given data collection to 0. Sample wise centre: this is a Boolean value that specifies that the mean for each individual sample should be set to 0. Feature wise stand normalisation: Using the standard that is established by the data set in a feature-wise fashion, a Boolean value indicates whether or not the input data should be divided. Sample wise standard normalisation: This is a Boolean value that divides each of the individual input values by standard. Zca epsilon: When left empty, this input takes on the default value of 1e-6, which is used to represent epsilon in ZCA whitening. ZCA whitening Boolean value that indicates whether or not the ZCA whitening should be applied. Rotation range: The range of degrees in random rotations is defined by this integer value. The width shift range parameter in the context of the tf.keras.preprocessing.image. ImageDataGenerator class serves as a key element to introduce variability into the training data through horizontal shifts. Its value can be specified as a floating point number or an integer. The range of allowed values for integers is within [-width shift range, +width shift range]. If the floating point number is less than 1, it represents a portion of the total width, allowing for smooth adjustments. If the floating point value is 1 or greater, it is interpreted as the number of pixels to move. Height shift range: Similar to the width sift range, it can be a float, int, or 1-dimensional array, but in this instance, it refers to the entire height. Split validation: It is a float value that specifies the fraction of photos that are utilised in the validation process. It should have a value between 0 and 1. Rescale: It stands for the rescaling factor, which might have a value of either channels first or channels last. (23)

Table 1 important Softwares and Tools

|  |  |
| --- | --- |
| Description | tools |
| Integrated Development platform | Pycharm and Visual Studio code(VSCode) |
| Programming Language | Python |
| Version Control | Git and Github |
| Python Libraries and Modules | Tensorflow,  pandas,  notebook numpy matplotlib seaborn python-box==6.0.2 pyYAML tqdm ensure==1.0.2 joblib types-PyYAML scipy Flask Flask-Cors |

# 3.2 RESEARCH DESIGN

# 3.2.1 TYPE OF RESEARCH

This study adopted an exploratory research design, focusing on qualitative analysis to understand the variations in the output of the VGG-16 model when images of different qualities were uploaded for the same image.

# 3.2.2 VARIABLES

Independent variables: Image quality (manipulated by adjusting resolution and compression).

Dependent variable: Output classification result of VGG-16 model.

# 3.2.3 DATA COLLECTION METHODS

Image Quality Adjustment: data for the study is created by adjusting the quality (in terms of percentage of quality) using the windows image editor. After this adjustments the copies of the same image with different memory size is saved and it can be use when predicting the image class.

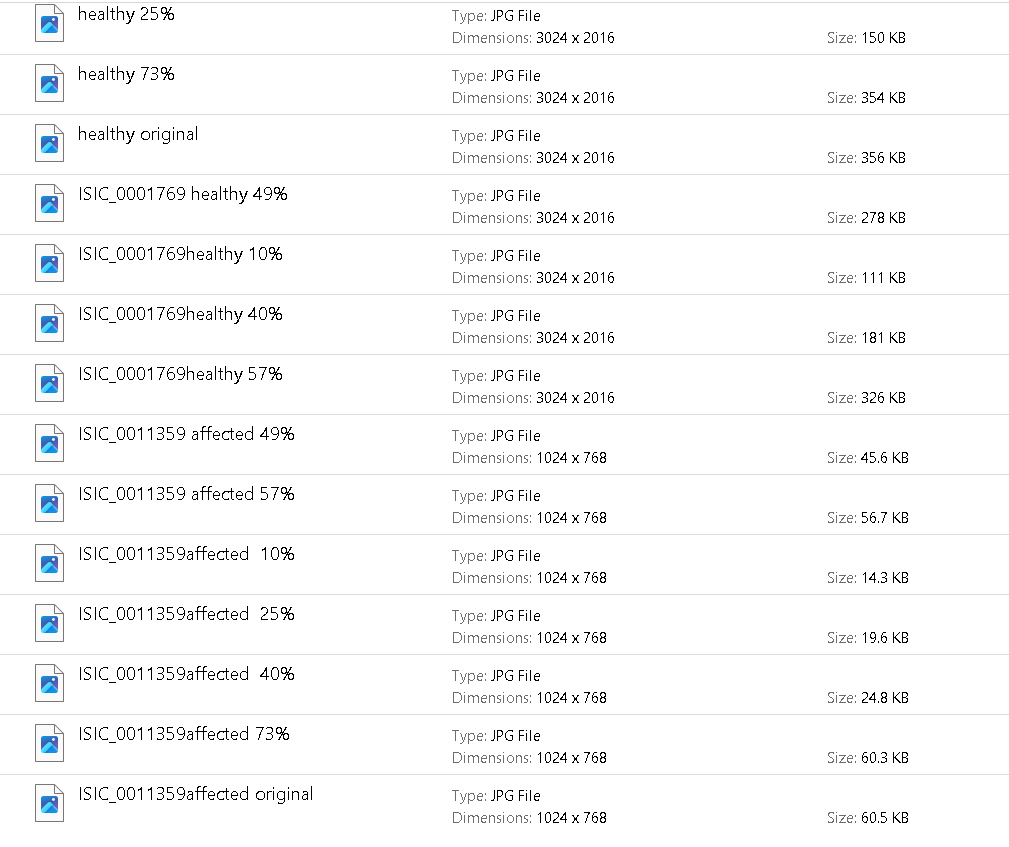


Figure 15 input images for the study

# 3.2.5 ANALYSES METHODS

After launching the web app, by running python file web app.py, each of the prepared images is uploaded to the web app. The results of each prediction is identified for getting a conclusion.

TOOLS AND TECHNOLOGIES

# Chapter : 4

# 4.0 RESULTS AND DISCUSSION

The study is conducted with the help of an AI based skin classification system, which can classify skin image to disease affected and normal skin. The main purpose of the classification system is to find the answer of the research question, that is the variation in the result prediction of a vgg16 based image classification system, when the quality of image is changed in terms of memory. The main area of study was the general change in prediction result in accordance with variation in image quality.

Table 2 results

|  |  |  |
| --- | --- | --- |
| Affected Skin Images | | |
| Original image 1 | Memory occupied | prediction result |
| quality reduced to 73% | 60.3 kb | no change |
| quality reduced to 57% | 56.7 kb | no change |
| quality reduced to 49% | 45.6kb | no change |
| quality reduced to 40% | 24.8 kb | no change |
| quality reduced to 25% | 19.6kb | no change |
| quality reduced to 10% | 14.3 kb | changed |

Table 3 results

|  |  |  |
| --- | --- | --- |
| Healthy Skin Images | | |
| Original image 2 | Memory occupied | Prediction result |
| quality reduced to 73% | 354 kb | no change |
| quality reduced to 57% | 326 kb | no change |
| quality reduced to 49% | 278 kb | No change |
| quality reduced to 40% | 278 kb | no change |
| quality reduced to 25% | 150 kb | no change |
| quality reduced to 10% | 111 kb | changed |

The outcomes demonstrate that the skin classification system based on VGG16 is generally capable of handling differences in image quality, as demonstrated by consistent prediction results under various scenarios. When the image quality is changed from the original quality to 73% of its original quality the memory is changed to 354 kb while the prediction result remains as healthy for the healthy skin image. As the image quality is reduced further to 57%, 49 % and 40%, the prediction result remained same. Followed by the quality reduction to 25 %( considerable decrease in memory to 150 kb), the prediction result changed at 10% of quality. In the case of the other class of image the prediction result remains same. In the case of affected skin images the memory decreased considerably, when the image quality is changed from 40% to 10%. However, the prediction result changed when image quality is reduced to 10%. This suggests that very low quality images can affect the model's ability to accurately classify skin conditions. Further research is needed to understand the specific challenges posed by low-quality images and identify possible strategies to improve model robustness in such scenarios.

# Chapter: 5

# 5: CONCLUSION

In this study, the variation in prediction results of an image classification system based on a pre trained vgg16 model is analysed. The skin classification system demonstrated overall consistency in prediction results to the changes in image quality. The results show that the model maintained accurate predictions for healthy skin images even when the image quality was reduced. However, for images of affected skin, reducing the image quality to 10% resulted in a significant change in the predicted results. This means that very low-quality images can affect the model's accuracy in classifying skin diseases.

Additionally, this study used VGG16 models implemented in TensorFlow and Keras and leveraged their features for efficient deep learning tasks. The images collected from ISIC data set is pre-processed and spitted automatically using image data generator tool. This data is utilized for the training and testing of the model. Transfer learning concepts are utilized in this study for creating an efficient image classification system using a small data set. The trained model achieved 79% with a loss of 3.329.

In summary, the VGG16-based skin classification system showed robustness to image quality variations for healthy skin images, but significantly affected the prediction results for low-quality images.This study provides insight into the challenges associated with image quality in dermatology AI models and paves the way for further investigation and improvement in this field.

PROJECT ARTEFACT: The complete implementation of the project is available on the following link: <https://github.com/ALBERTMICHAEL7/skin-disease-classification-system>

One drive link- [submission files, com726, Albert Pulickeel Michael](https://ssu-my.sharepoint.com/:f:/g/personal/6micha94_solent_ac_uk/EmGibNZ3QyFIi37rLpQy7Y0BM8Kb_ihX05XaKb-MFvDMLw?email=kashif.talpur%40solent.ac.uk&e=5Vsbd7)

# REFERENCE

REFERENCES

1. Yadav, N., Kumar, V. and Shrivastava, U. (2016). Skin Diseases Detection Models using Image Processing: A Survey. *International Journal of Computer Applications*, 137(12), pp.34–39. doi:https://doi.org/10.5120/ijca2016909001.
2. 3) American Academy of Dermatology Association (2019). *Skin conditions by the numbers*. [Online] Aad.org. Available at: <https://www.aad.org/media/stats-numbers>.
3. ) A.A.L.C. Amarathunga, E.P.W.C. Ellawala, G.N. Abeysekara, C. R. J. Amalraj Expert System For Diagnosis Of Skin Diseases INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH VOLUME 4, ISSUE 01, JANUARY 2015 ISSN 2277-8616.

4) Fu’adah, Y.N., Pratiwi, N.C., Pramudito, M.A. and Ibrahim, N. (2020). Convolutional Neural Network (CNN) for Automatic Skin Cancer Classification System. *IOP Conference Series: Materials Science and Engineering*, 982, p.012005. doi:https://doi.org/10.1088/1757-899x/982/1/012005.

5) Better Health Channel (2014). *Melanoma*. [Online] Vic.gov.au. Available at: https://www.betterhealth.vic.gov.au/health/conditionsandtreatments/melanoma.

6) Rosen, G. (2015). *A history of public health*. Baltimore: Johns Hopkins University Press.

7) Kim, I., Shim, H. and Yang, J. (n.d.). *Face detection*. [Online] Available at: https://web.stanford.edu/class/ee368/Project\_03/Project/reports/ee368group02.pdf.

8) Kleinman, Z. (2021). Google AI tool can help patients identify skin conditions. *BBC News*. [Online] 18 May. Available at: https://www.bbc.co.uk/news/technology-57157566.

9) SkinVision. (2017). *Skin Cancer | Complete Overview of Symptoms & Types*. [Online] Available at: https://www.skinvision.com/skin-cancer/.

10) IBM (2023). *What is Machine Learning?* [Online] www.ibm.com. Available at: https://www.ibm.com/topics/machine-learning.

11) Fatima, M. and Pasha, M. (2017). Survey of Machine Learning Algorithms for Disease Diagnostic. *Journal of Intelligent Learning Systems and Applications*, 09(01), pp.1–16. doi:https://doi.org/10.4236/jilsa.2017.91001.

12) M.S. Poornima, K. Shailaja, Detection of Skin Cancer Using SVM, Int. Res. J. Eng. Technol. 04 (2017). Vol. 4, July 2017.

13) Rathod, J., Waghmode, V., Sodha, A. and Bhavathankar, P. (2018) Diagnosis of Skin Diseases Using Convolutional Neural Networks. Proceedings of the 2nd International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, 29-31 March 2018, Vol. 2, 1048-1051.  
<https://doi.org/10.1109/ICECA.2018.8474593>

14 Simonyan, Karen, and Andrew Zisserman. VERY DEEP CONVOLUTIONAL NETWORKS for LARGE-SCALE IMAGE RECOGNITION. 2015. [Online] arXiv.org. Available at: https://arxiv.org/abs/1409.1556v6.

15) Tammina, S. (2019). Transfer learning using VGG-16 with Deep Convolutional Neural Network for Classifying Images. *International Journal of Scientific and Research Publications (IJSRP)*, 9(10), p.p9420. doi:https://doi.org/10.29322/ijsrp.9.10.2019.p9420.

(<http://www.ijsrp.org/research-paper-1019.php?rp=P949194#citation>)

16) Sudha, M. and Poorva, B. (2019). Predictive Tool for Dermatology Disease Diagnosis using Machine Learning Techniques. *International Journal of Innovative Technology and Exploring Engineering*, 8(9), pp.355–360. doi:https://doi.org/10.35940/ijitee.g5376.078919.

17) Akyeramfo-Sam, S., Addo Philip, A., Yeboah, D., Nartey, N.C. and Kofi Nti, I. (2019). A Web-Based Skin Disease Diagnosis Using Convolutional Neural Networks. *International Journal of Information Technology and Computer Science*, 11(11), pp.54–60. doi:https://doi.org/10.5815/ijitcs.2019.11.06.

18) Yusof, M.M., Aziz, R.A. and Fei, C.S. (2013). The Development of Online Children Skin Diseases Diagnosis System. *International Journal of Information and Education Technology*, 3(2), pp.231–234. doi:https://doi.org/10.7763/ijiet.2013.v3.270.

19) N3)TensorFlow 2019, TensorFlow, TensorFlow, Google

20) N4) https://www.databricks.com/glossary/tensorflow-guide

21) Memorial Sloan Kettering Cancer Center n.d., *ISIC Archive*, in DA Scope (ed.), [www.isic-archive.com](http://www.isic-archive.com).

22) Udrea, A, Mitra, GD, Costea, D, Noels, EC, Wakkee, M, Siegel, DM, Carvalho, TM & Nijsten, TEC 2019, ‘Accuracy of a smartphone application for triage of skin lesions based on machine learning algorithms’, *Journal of the European Academy of Dermatology and Venereology*, vol. 34, no. 3, pp. 648–655.

23) n5 EDUCBA 2022, *Keras ImageDataGenerator | What is keras ImageDataGenerator?*, EDUCBA.

24) n6. Otto, M 2022, *Bootstrap*, Getbootstrap.com.

25) Amidi , A & Amidi, S 2012, *The evolution of image classification explained*, Stanford.edu. (n7)

26) ALKolifi ALEnezi, NS 2019, ‘A Method Of Skin Disease Detection Using Image Processing And Machine Learning’, *Procedia Computer Science*, vol. 163, no. 1877-0509, pp. 85–92, viewed 21 May 2020, <https://www.sciencedirect.com/science/article/pii/S1877050919321295>. (n8)

27) Guan, Q, Wang, Y, Ping, B, Li, D, Du, J, Qin, Y, Lu, H, Wan, X & Xiang, J 2019, ‘Deep convolutional neural network VGG-16 model for differential diagnosing of papillary thyroid carcinomas in cytological images: a pilot study’, Journal of Cancer, vol. 10, no. 20, pp. 4876–4882. ()

28) Memorial Sloan Kettering Cancer Center n.d., *ISIC Archive*, in DA Scope (ed.), [www.isic-archive.com](http://www.isic-archive.com).

# APPENDICES

APPENDIX A:

## ETHICAL CLEARANCE

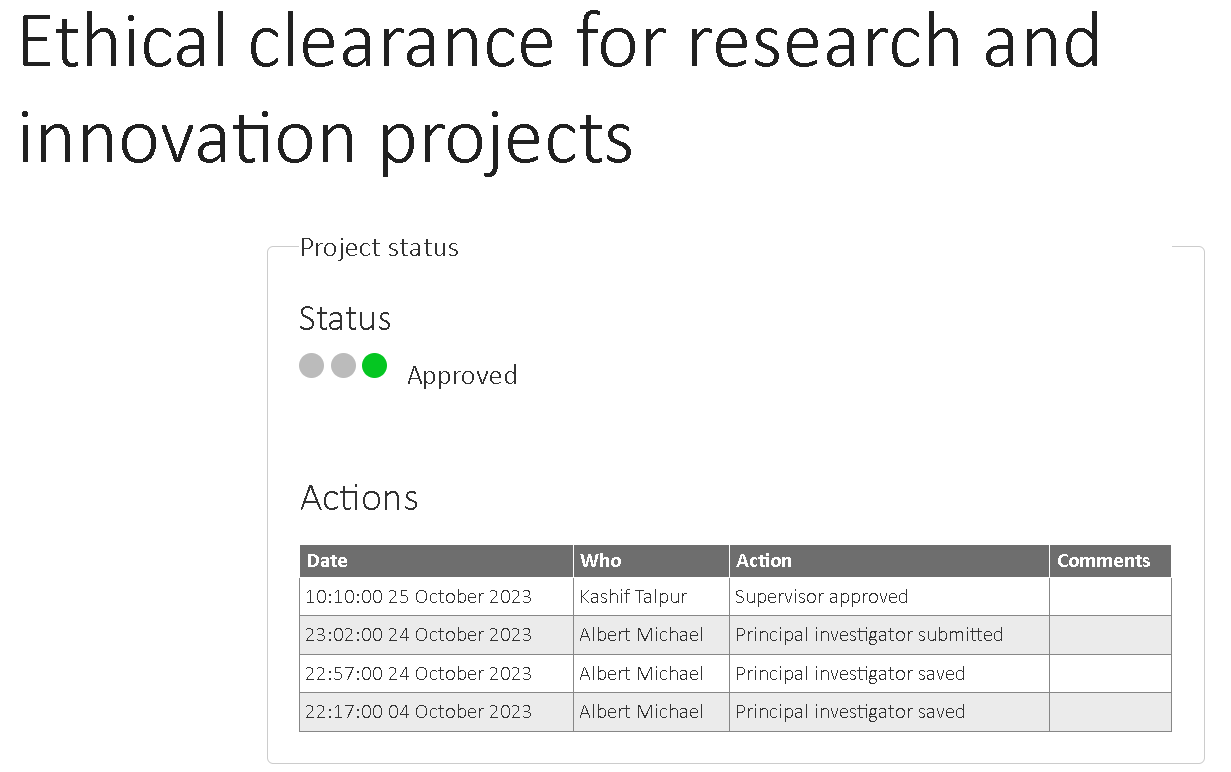


Figure 16- ethical clearance

APPENDIX B:

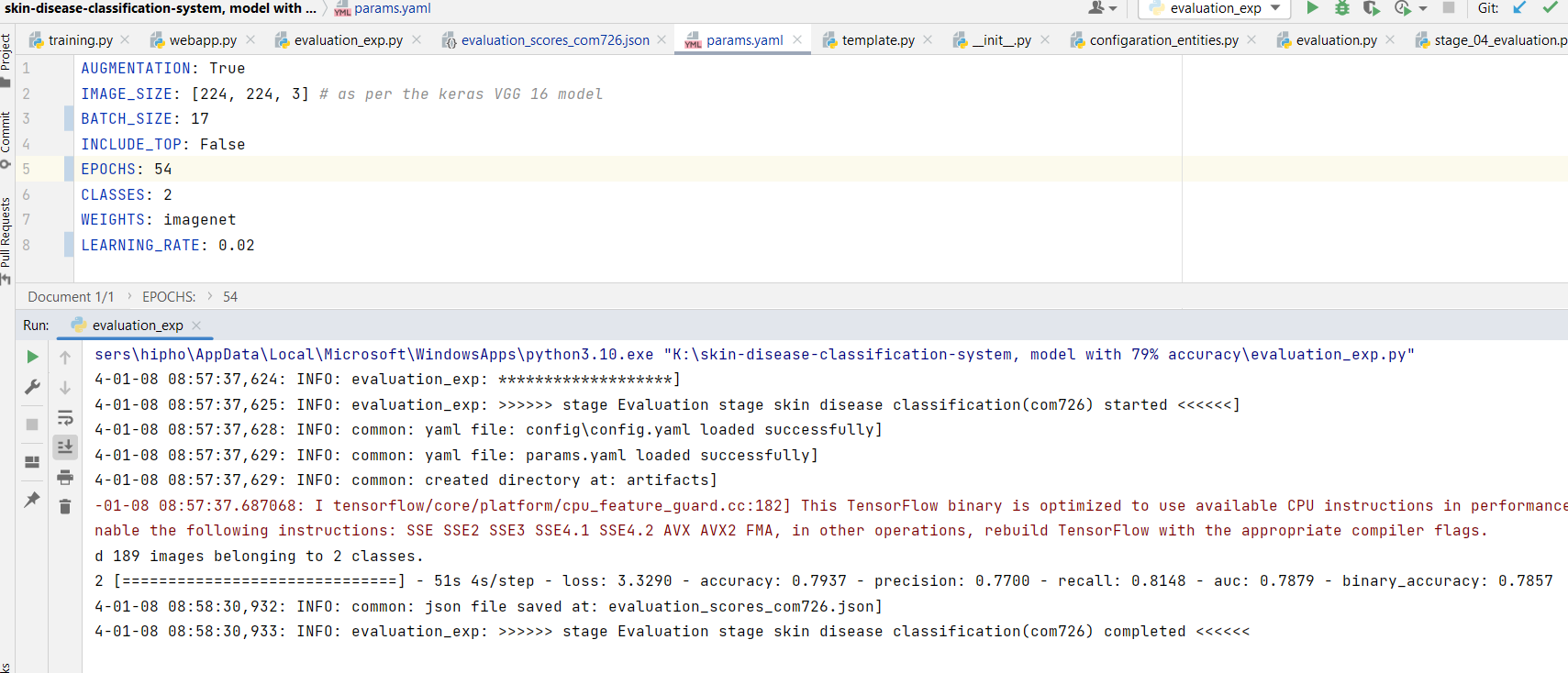


Figure 17 hyper parameters and model accuracy for that parametrs

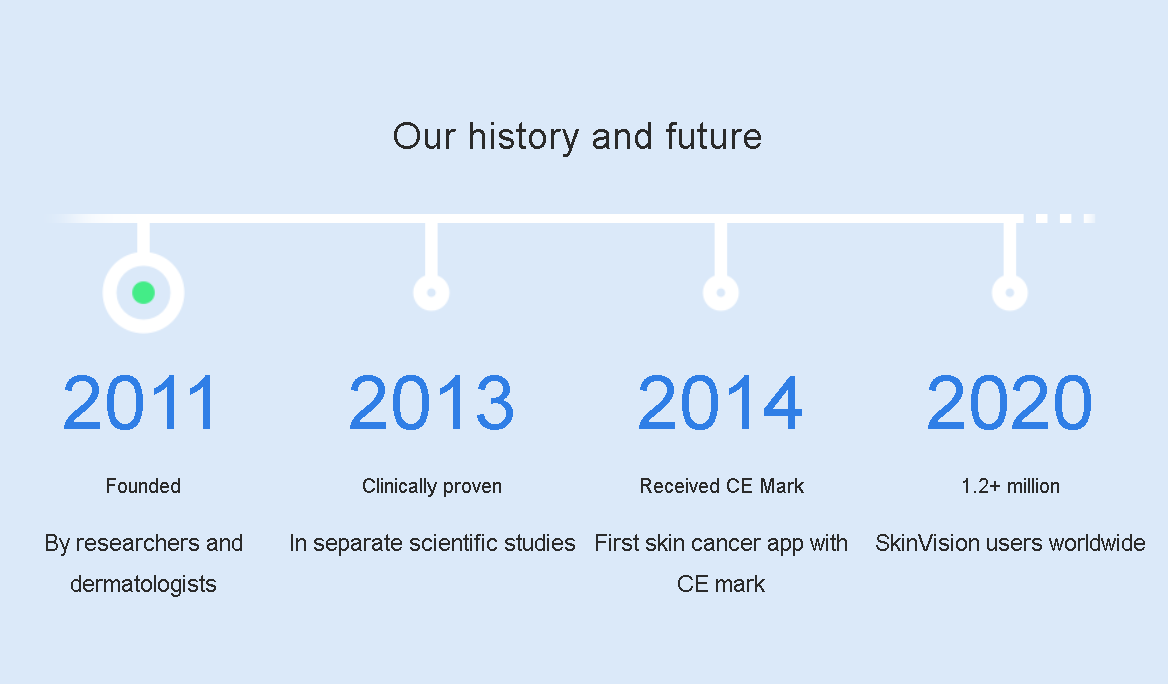


Figure 18 skin vision app