

Computer Vision Based Skin Disease Detection Using Machine Learning

by

Ahmad Wasiq Jayeb

18101057

Alvin Rahul Hore

18101326

Ramisa Anjum

18101206

Sohana Sanjana Sadeque

18101612

Syed Tahsin Auqib

18101488

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

Department of Computer Science and Engineering
Brac University
September 2022

© 2022. Brac University
All rights reserved.

Declaration

It is hereby declared that

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


Student's Full Name & Signature:



Ahmad Wasiq Jayeb
18101057



Alvin Rahul Hore
18101326



Ramisa Anjum
18101206



Sohana Sanjana Sadeque
18101612



Syed Tahsin Auqib
18101488

Approval

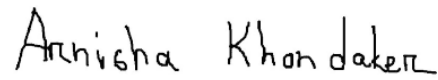
The thesis/project titled “Computer vision based skin disease detection using machine learning ” submitted by

1. Ahmad Wasiaq Jayeb (18101057)
2. Alvin Rahul Hore (18101326)
3. Ramisa Anjum (18101206)
4. Sohana Sanjana Sadeque (18101612)
5. Syed Tahsin Auqib (18101488)

Of Summer, 2022 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on September, 2022.

Examining Committee:

Supervisor:
(Member)



Arnisha Khondaker
Senior Lecturer
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Md. Golam Rabiul Alam, PhD
Designation
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

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

Abstract

Skin cancer have been the primary focus of this study, as they are one of the most deadly diseases if not diagnosed and treated early. The study will make it possible for computer science and medical science to work together to save lives. Machine Learning, Deep Learning, and Image Processing have already been used to treat skin conditions. Even though Deep Learning, Machine Learning, and Image Processing have all been used in the past to treat skin diseases, we are trying to improve the accuracy of this work by implementing new Image Processing and Machine Learning models. In this study, pre-trained CNN models and combined pre-trained Unet models are used to identify skin cancer and categorize the kind of cancer using datasets with labeled pictures and classes. Although it may develop in areas of your skin that are not exposed to sunlight, skin cancer is an abnormal growth of skin cells that most frequently affects skin that has been exposed to the sun. Skin cancer is a kind of malignant melanoma, which is a type of cancer. The three most common kinds of skin cancer are melanoma, basal cell carcinoma, and squamous cell carcinoma. Approximately 301 people die from skin cancer each year. It will be difficult to achieve high accuracy if you rely just on the dataset provided by Kaggle. Recognize that not all datasets are balanced. Therefore, this study focuses on identifying several methods to combine Deep CNN models with U-net to produce deep CNN models that have the highest accuracy on both large and small datasets. These methods mostly depend on supervised learning, which involves the use labels and training data from datasets. This research aims to demonstrate how the best model can be used to accurately diagnose Skin cancer and disorders at an early stage. Here, we have classified a unique dataset using U-net model. We further examined the CNN, U-net, pre-trained models, and other Unet based models for example Attention Unet, ResUnet, R2Unet models accuracy to develop an optimum model that may be further customized to a mobile application for widespread usage. On more than 10000 photos of seven different skin conditions compared to healthy skin, we built the architecture. On the basis of the images of our researched dataset, we carefully compared our data and classified it. Finally, using ResUnet, which is the best model for the task at hand, we accurately identified the Seven illnesses with a 87.21% accuracy rate.

Keywords: Image Processing; Machine Learning; Deep Learning; Epoch; SoftMax; Skin Disease; Skin Cancer; KNN; Classification; CNN; U-net; Detection; Accuracy; Validation; TensorFlow; Keras; Keras Layer; Dense Layer; ResUnet; R2Unet; Attention Unet .

Dedication

Our thesis is dedicated to my institution's supervisors, under whose persistent supervision We finished this dissertation. They not only educated us with academic knowledge, but they also provided us with helpful guidance when We needed it the most. Again, We would want to dedicate our efforts to our parents. We also like to thank Almighty for completing our work successfully.

Acknowledgement

First and foremost, we thank the All-Powerful Allah for enabling us to complete our thesis without experiencing any major hurdles.

Second, we want to thank our adviser, Ms Arnisha Khondaker, a lecturer at BRAC University, for her invaluable guidance and assistance with the thesis and report writing, as well as her enthusiasm, guidance, and immense support. We have been inspired and motivated by her guidance, incorporation, and involvement throughout our research. She never hesitated to help us out when we needed it.

Thirdly, we might not be able to accomplish our goals without the constant support of our parents. Without their generous help and prayers, we wouldn't be close to graduation. All of the individuals named here deserve our gratitude.

And finally Last but not least, we would want to express our gratitude to the faculty, seniors, friends who obliquely encouraged and inspired us through their research. We also acknowledge the help we received from various online sites, particularly from the research of our fellow researchers.

Table of Contents

| | |
|---|-----------|
| Declaration | i |
| Approval | ii |
| Abstract | iii |
| Dedication | iv |
| Acknowledgment | v |
| Table of Contents | vi |
| List of Tables | viii |
| Nomenclature | ix |
| 1 Introduction | 1 |
| 1.1 Motivation | 2 |
| 1.2 Research Problems | 3 |
| 1.3 Research Objectives | 4 |
| 2 Literature Review | 5 |
| 2.1 Related works | 7 |
| 2.2 Categories of skin cancer | 10 |
| 3 Dataset Descripton | 20 |
| 3.1 Dataset Background | 21 |
| 3.2 Classification of Dataset | 22 |
| 3.3 Technical Validation | 24 |
| 3.4 Technical Terms and Definitions | 24 |
| 4 Pre-processing of Dataset | 26 |
| 4.1 Libraries Required | 27 |
| 4.2 Cleaning of Data | 28 |
| 4.3 Exploratory Data Analysis | 28 |
| 5 Methodology | 31 |
| 5.1 Implementation of CNN | 31 |
| 5.2 Implementation of Unet | 35 |
| 5.3 Implementation of Attention U-net | 37 |

| | | |
|----------|---|-----------|
| 5.4 | Implementation of ResUnet | 38 |
| 5.5 | Implementation of R2U-net | 41 |
| 6 | Model Building | 43 |
| 6.1 | Introduction to CNN | 43 |
| 6.2 | Introduction To U-Net | 45 |
| 6.3 | Introduction to Attention U-net | 49 |
| 6.4 | Introduction to ResUNET | 51 |
| 6.5 | Introduction to R2U-net | 51 |
| 7 | Model Evaluation and Result Analysis | 53 |
| 7.1 | Fitting Model | 53 |
| 7.2 | Transfer Learning Using One Pre-trained Model | 53 |
| 7.3 | Analysis and Comparison of the Models | 58 |
| 8 | Conclusion | 60 |
| | Bibliography | 63 |

List of Tables

| | | |
|-----|---|----|
| 3.1 | Overview of ISIC archive and Class Distribution upto April 2020 . . . | 22 |
| 7.1 | Test accuracy and loss Comparison of all models | 58 |

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

ANN Artificial Neural Network

bcc Basal cell Carcinoma

bkl Benign Keratosis

CNN Convolutional Neural Network

df Dermatofibroma

EH Edge Histogram

HAM Humans Against Machine

Mel Melanoma

NN Neural Network

nv Melanocytic Nevi

RGB Red, Green, Blue

Vase Vascular Skin Lesions

Chapter 1

Introduction

The term "cancer" is used to refer to a group of disorders that are caused by irregular cell growth and have the potential to spread to other parts of the body. Cancer virtually exclusively starts in cells. One cell or a very small number of cells could show the first indications of malignancy. The World Health Organization (WHO) claims that it will account for over 10 million deaths in 2020, or almost one death for every six, making it the leading cause of death worldwide. According to research, prompt and precise diagnoses can significantly lower the cancer-related death rates. Even though accurate diagnoses or false negatives are frequently made, there are situations when a high level of knowledge is required. It is essential to do research into the creation of a computerized diagnostic system that is supported by deep learning due to the inherent fallibility of human diagnosis. We demonstrate a system that is capable of detecting the various types of skin lesion by utilizing a deep learning method - CNN and U-net based models.

Skin cancer, or the uncontrolled proliferation of skin cells, is more inclined to occur on sun-exposed skin. In addition, this prevalent type of cancer can also develop in skin tissue that is not frequently exposed to light. Skin cancer is a general word that refers to two distinct types: melanoma and non-melanoma skin cancers. Melanoma, which was recently ranked as the 19th most common disease, was discovered to be the most common variety. Approximately 300,000 new instances of cancer were detected in 2018. More than a million people are affected by non-melanoma cancer, which is the fifth most prevalent malignancy.[9]

Due to a variety of reasons, estimating the incidence of skin cancer poses a special set of difficulties. Accurate data on skin cancer might be challenging to get because there are many different types of the disease. For instance, cancer registries typically ignore non-skin cancers like melanoma. The worst type of skin cancer is melanoma. Because the large proportion of cancer cases may be treated with surgery or excision, this record is typically erroneous.

A subfield of machine learning is called deep learning. Convolutional Neural Networks act as the "brain" of the system, allowing it to work. The lack of a powerful

personal computer in the past made it challenging for us to supply the required data for the training of enormous neural networks. A convolutional neural network is one form of deep learning method (CNN). Many different layers are routinely used by convolutional neural networks. for the purpose of retrieving image information required for object classification. These features are extracted by CNN kernels. Large pixel discrepancies are found by the edge kernel. It evaluates a picture and emphasizes recognizable shapes. Compared to other networks, CNN needs minimal preprocessing. CNN has a benefit over its predecessors because it can identify important aspects without user input and because its structure resembles the neural connections seen in the human brain. With the help of a convolutional neural network, we can now identify skin cancers at an earlier stage.[23] This might be achieved by giving the algorithms a substantial amount of information. When given an image, the training algorithm evaluates it in accordance with its directions and searches within its own learning algorithm for a specimen of skin cancer. The ability to detect the precise type of skin cancer will make it easier for patients to receive curative therapy at an earlier stage if the results are consistent with those of the trained model.

U-net is also a type of technique for deep learning. The U-Net is a sophisticated design that takes care of the bulk of issues that crop up. This approach benefits from the concept of fully convolutional networks. The localization and context features are to be recorded by the U-Net. The kind of architecture that was constructed successfully completes this procedure. The fundamental idea behind the implementation is to use upsampling operators directly after successive contracting layers in order to produce outputs with greater resolution on the input images. It is used for image segmentation. In image segmentation when we segment an image, we break it up into smaller pieces. These more manageable pieces will aid in the computation of the image segmentation task.[28]

1.1 Motivation

Among vital organs in the human body, skin provides the strongest level of protection. Our inside organs are shielded from injury by this shield. However, this critical component of the infections that are so severe they harm the human body include a little dust, fungus, virus, etc. All around the world, a variety of skin conditions affect millions of people. Skin disorders are becoming more prevalent worldwide, and these illnesses are what are causing an increase in skin cancer cases. By looking at the skin scans, it can be found and, if feasible, treated early. We can therefore anticipate whether it will be fatal or not. But occasionally, something does not go unnoticed, and by that time it is too late for him. Additionally, receiving care through the healthcare system is not always an option because it can be expensive or cause the process to start much later than expected. By performing a biopsy, which entails removing a small sample of tissue from the patient's body for examination, doctors can determine the diagnosis. Similar to how X-rays can identify a digital representation of an object's internal structure, particularly a section of the body. But the only option to treat skin cancer is through a painful operation called a biopsy, which may also leave patients susceptible to infection or bleeding.

The biopsy procedure takes a significant amount of time, requires a lot of medical expertise, and occasionally the results are erroneous. The diagnosing patient could pass away due to complications. Therefore, in order to treat the population, we need more highly skilled doctors due to this immense complexity. Unfortunately, patients have more medical knowledge than doctors do. As a result, doctors are currently looking into machine learning algorithms as alternative options.[20]

Because image processing can both save doctors a ton of time and improve research efficiency, this work inspired us to consider the mechanisms for diagnosing disease or cancer and how technology can expedite the process for the medical professionals. So in this paper we use Convolution Neural Network and Unet segmentation on the images.

1.2 Research Problems

The best way to diagnose and treat an illness is by consulting a doctor, but as technology has advanced, we are now more aware of the application of Deep Learning in the medical industry. As a result, various models can be employed for image classification, however finding and implementing the method that best fits our model was challenging. Inadequate datasets or lesser levels of accuracy were present in earlier research publications in the same topic. A comparison of various models to obtain the best outcome was not apparent in the earlier research, and the appropriate models were also lacking. More research was therefore necessary to at least clear the way for future projects. Therefore, this study did not just produce the best model, both the Unsupervised Approach and customized big datasets were used. The most difficult aspect for the entire project is image processing, which involves extracting each pixel from a specific image to match with a certain dataset using deep learning and identifying appropriate labels. Any action, however, cannot identify the impacted area from an image without a sufficient dataset that has the relevant titles. A dataset's validation is important once more. A dataset without sufficient authentication could cause the entire model to produce an incorrect result. The most important component of this model is thus choosing an actual, reliable, and genuine dataset. In machine learning, the term "data bias" refers to a mistake where some types of databases are more likely to occur and may seriously impair the entire model, ultimately resulting in its failure. When a dataset contains a certain quantity of data multiple times, the computer uses the data and becomes biased toward those numbers, which is what causes the repeated data to provide the desired result. Repeated data creates a large bias in a dataset, which is responsible for the model's overfitting. An overfit model, on the other hand, is an illustration of a complex model that renders the entire model inefficient and ineffective.[3] Understanding all of the medical jargon used to describe skin illnesses in order to identify them in the dataset is an extremely difficult and hard task. Again, the overall lack of knowledge of several other languages makes it difficult to comprehend and act appropriately. Medical picture segmentation can be effectively accomplished using convolutional neural networks (CNN), particularly the Unet. Unset has so far shown state-of-the-art performance in a variety of challenging medical image segmentation tasks, particularly when the distribution of the training and testing data is the same (i.e.

come from the same source domain). But in actual clinical settings, several suppliers and facilities provide the medical images. When transferred to a new target domain (for example, a different vendor or acquisition parameter), the performance of a U-Net learned from a certain source domain may unexpectedly decline. It is expensive, time-consuming, and nearly hard to retrain the U-Net by gathering a significant quantity of annotation from each new domain.

1.3 Research Objectives

Our objective is to develop a model that offers us the correctness of the skin cancer in order to avoid the painful medical procedure of biopsy. If it can tell us what kind of skin condition it has, that would be an added bonus. This study aims to improve the accuracy and speed of the identification, classification and segmentation of seven different skin conditions (melanoma, acne, basal cell carcinoma, actinic keratosis, viral skin infections, bacterial skin infections, deep dermal problem, tinea, eczema). The study also successfully categorizes the new skin to more precisely identify the disorders. In order for the software to determine whether a skin has a disease or is healthy and free of any ailment, the Fresh skin dataset was implemented with the intention of being more disease-focused. Deep learning learns from the raw data and automatically classifies after learning to identify the precise, useable information from images. The seven different types of skin cancer will be categorized in this article. Deep CNN and U-net's functioning and effectiveness will also be assessed and examined. The main goal is to create a model that uses the Convolution Neural Network to identify cases of skin cancer and classify them into several categories and U-net to record the features of the skin and localization as well. Our objectives includes –

1. Clearly identifying the damaged skin cells and the damage-causing mechanism.
2. Identify the variations between skin that is sick and skin that is not.
3. Presenting each disease's categorization and likelihood as a percentage, with the disease with the highest percentage being taken into consideration as the potential cause of the condition.
4. Improving a disease detection, classification and segmentation model.
5. Assessing the model's methodology and algorithm.
6. Supplying the essential suggestions for improving the model.
7. Processing the model in a way that only the afflicted cells can be detected while taking into account the skin tone variations.
8. Finally comparing many models to achieve the best outcomes.

Chapter 2

Literature Review

The term "Convolutional Neural Network" is abbreviated as "CNN." The phrase "convolutional neural network" refers to a network that uses a convolutional system of linear equations to solve problems. This kind of neural network uses matrix convolution instead of the more basic process of matrix multiplication in at least one of its layers. Convolutional neural networks, also referred to as CNN or ConvNet, are a particular kind of artificial neural network that deep learning uses to evaluate visual information. CNN is a kind of deep learning method for processing data using an automatically formed, adaptive grid pattern. CNN created this pattern, which was informed by the structure of the animal visual brain. This layered structure consists of an input layer, a hidden layer, and an output layer. Its most common applications include image processing, classification, segmentation, and other auto-correlated data types. CNN has a significant advantage over its competitors as it can detect key characteristics without the involvement of a human. If it is shown a huge number of pictures of cats and dogs, it can figure out on its own the specific qualities that are unique to each class. Convolutional Neural Networks, for instance, start with an image and divide it into several layers before searching for matches using the inputted data and aiming to get the most maximally matched outcomes. Furthermore, CNN has a minor computational overhead.[18]

But there is no disputing that it has certain drawbacks as well. For example, it does not retain the location or orientation of the objects being evaluated and requires a significant amount of information for training.

A tensor of shape serves as the input to a CNN. A tensor is a mathematical concept related to, but much more general than, a vector. It has been characterized by a set of components that are functions of spatial coordinates. The picture is turned into an abstract form known as a feature map, which has shape, after passing through a convolutional layer. Because the input size of images is so vast and each pixel is such a significant attribute, even the most basic models would need an enormously high number of neurons. Following that, each neuron computes an output value by applying a preset function to the input values received from the layer below it in the hierarchy's receptive field.

Each input image will be processed by a sequence of convolution layers, including filters (Kernels), Pooling, fully connected layers (FC), and the Softmax function, in order to train and assess deep learning CNN models. The purpose of this procedure is to identify a picture with probabilities ranging from 0 to 1, and it will be done sequentially. CNN is just useful for viewing images in practice.

It may be added to any 2D or 3D array. The layers that are introduced to the neural network before the layout is finished are the convolutional layers. Convolution layers are used to enhance the computer's capacity to detect features that would otherwise be lost if an image was simply processed into pixel values. These properties would be lost if the image was converted to pixels.[5]

TensorFlow: An accessible library for artificial neural networks is called TensorFlow. It is a library for symbolic math that is built on data flow and differentiable programming. It offers a collection of techniques for creating and honing models in JavaScript or Python, as well as deploying them online, locally, through a browser, or on portable devices. You may use the tf.data API to mix modules to create huge input streams. The most crucial aspect of TensorFlow to understand is that its core is primarily not written in Python: It's written in a combination of C++ and CUDA for the greatest results. It's an open-source machine learning platform from start to finish. The open source TensorFlow framework allows for the building of highly flexible CNN architecture for computer vision challenges. TensorFlow is used for a wide range of tasks such as voice recognition, sentiment analysis, language identification, text summarization, image recognition, video detection, time series, and more. According to users and industry experts, TensorFlow is tough to understand and much more difficult to operate. TensorFlow is notorious for its lack of flexibility, yet research is all about flexibility, so mastering it is difficult. It may be used for a number of things other than deep learning. A client session object may be used by the caller driver of the tensor flow graph created using the C++ API. In a nutshell, TensorFlow simplifies machine learning. Anyone can now create ML models thanks to pre-trained models, data, and high-level APIs. Researchers often use the following terms: TensorFlow is used by the vast majority of researchers and students for analysis and model creation.

Keras: Keras is a free and open-source software program that provides a Python interface to an artificial neural network. It works with a variety of backends, including TensorFlow, Microsoft Cognitive Toolkit, Theano, and PlaidML. Francois Chollet, a Google developer, was the one who created and is still maintaining it. Keras is made up of numerous different types of layers. Examples include layers with names like "Dense layer," "Flattened layer," "Dropout layer," "Reshape layer," "Permute layer," "Lambda layer," and "Pooling layer." It is basically a comprehensive and user-friendly Python library that is open source and free and can be used for deep learning model building and assessment. Keras is a more user-friendly solution for developers than TensorFlow. For first-timers, learning keras is substantially easier. It is an application programming interface (API) created for humans rather than machines. It is based on a simple structure, which leads to the development of a

simple approach for generating deep learning models.

2.1 Related works

This particular section briefly discusses the associated initiatives and studies that have been conducted to identify research gaps. A Skin Disease Diagnosed by Smartphone A similar smartphone application called "Classification Using MobileNet CNN" uses the MobileNet model to mainly identify skin problems. On seven skin conditions, they used the transfer learning approach while using the MobileNet model. As a consequence, a method for classifying skin diseases was developed for an Android app. The authors collected 3,406 photos, and it was found that the classes had different numbers of photographs in them. Therefore, it was regarded as an unbalanced dataset. They used several preprocessing and sampling techniques to the raw data in order to increase accuracy. "Mobile Application for Preliminary Diagnosis of Disease" is another comparable piece of work that focuses on identifying common diseases and offering suggestions. This specific application includes an information system that can analyse the medical symptoms of the patient and provide a preliminary diagnosis. It may also suggest doctors that specialize in a certain field, which truly aids people in locating the medical care they need for their present situation.

The performance of six proposed techniques (Balanced Random Forest, Balanced Bagging, AdaBoost, Random Forest, Logistic Regression, Balanced Bagging SVM) was evaluated using 2,453 photos. The model was made using dermoscopic images of seven different skin conditions, including basal cell carcinoma, melanocytosis, and melanoma. In a recent study, Van-Dung Hoang, Chi-Mai Luong, Antoine Doucet, Cong-Thanh Tran Tri-Cong Pham stated that the EfficientNetB4-CLF model had the greatest accuracy (89.97%), lowest recall (86.13%), and fewest recalls (0.39%). They created data level strategies that use a minority dataset percentage until the proportion matches the maximum classes. Better than EfficientNetB4 in image classification was DenseNet169. The CNNs come from the trained network of ImageNet. Every hidden layer in Custom Fully Connected Layers (CFCL) comes after a 0.2 dropout block to prevent overfitting. Their research used Adam Optimizer to improve their network in the following ways: Beta 2 = 0.999, lr is dynamic between 0.000001 and 0.00005, and amsgrad = Not True. for changing the LR one epoch at a time, and CyclicLR. A method for raising a CNN architecture's weight values is called back-propagation. In this research, the scientists utilized 24,530 pictures. These images have been reduced in size to 256*192 pixels. The images were divided into three categories: test, train, and validation; test represented 10% of the images, train 80%, and validation 10%.[4]

Additionally, Neha Agrawal and Dr. Sagaya Aurelia published a research paper on strengthening the medical sector by enhancing the diagnosis of three illnesses (Vascular Tumors, Vitiligo, and Melanoma) via the use of transfer learning. They used a transfer learning model by the name of InceptionV3. Accuracy increases when a large dataset was being used. They utilized Kaggle data at the time of writing and

discovered datasets of skin infection that were available for 117 categories. In reality, Keras was used to build the main models, while NumPy and Matplotlib were used to graph the accuracy and loss. The Adam optimizer has a learning rate of 0.001. There were 34,432 non-trainable parameters and 21,808,931 trainable parameters used here. 10 layers and sixteen clusters were used to train the Inception V3 model over the course of 27 batches and ten epochs. Categorical cross-entropy served as the loss function for the Adam optimizer. Although test and pre-training accuracy were both 81.70% and 84.84%, respectively, fine-tuning boosted detection accuracy by 12.79%.[1]

The extents of a novel architecture known as DRANet were described in a publication titled A Visually Interpretable Deep Learning Framework for Histopathological Image-Based Skin Cancer Diagnosis by several IEEE members. The DRANet deep learning system suggests to differentiate eleven distinct types of skin illnesses based on a real histological picture that was amassed during the previous ten years. They suggest using CAM, or Class Activation Maps, to visualize deep neural networks. DRANet is particularly useful because to its tiny model size and great classification accuracy. The model is 86.8% accurate overall, 88.8% accurate on a weighted average, 86.8% accurate on callback, and 87.1% accurate on F1 for eleven epidermal circumstances. RANet and DRANet-BN perform better on average by 3.8%, 5.9%, 3.8%, and 4.4%, respectively. EfficientNet-B0, ShuffleNet V2, MobileNet, NAS-Net Mobile, and MobileNetV2 are all outperformed by DRANet (86.8% vs. 75.6%, 78.2%, 81.6%, 77.8%, and 78.6%). On all metrics, DRANet performs better than EfficientNetB1 with the exception of F1 score (86.8% vs. 86.5%) and recall (86.8% vs. 86.3%). (4084K versus 6589K). With 86.8% accuracy, DRANet is somewhat more accurate than 82.1% of VGG19 82.1% of VGG16, and is comparable to InceptionV3 (86.3%) ResNet50 (85.5%).

Zhang X, Wang S, Liu J, and others According to a research, deep learning algorithms were used in the study to help in the identification of recurring impacted skin illness predicated on dermoscopy photos. This was done in an effort to improve the diagnosis of skin disorders by fusing deep neural networks human expertise. Their system's accuracy on a test dataset of 1067 images was 87.25 2.24%. The following conditions were selected for a prior study: Seborrheic Keratosis (SK), Melanocytic nevus, Psoriasis, and Basal Cell Carcinoma (BCC). They developed their solution using GoogleNet InceptionV3's code package, which had been pre-trained on around 1 million pictures.They used subcategories inside categories to identify the errors. Additionally, their findings show a standard deviation that ranges between two and five percent, which may indicate accuracy variation. The goal is to create a user-friendly decision support system that is available through mobile apps for patients and will help professionals make more accurate diagnostic judgments.[24]

Albawi, Saad et al. published Robust skin disorders detection and classification using deep neural networks. Melanoma, nevus, and atypical identification all used the ISIC dataset. Other methods for extracting ROI include statistical region merging,

iterative region merging, and adaptive threshold. The strategy employs average and Gaussian filters to accurately predict skin diseases. Even before to feature extraction and classification operations, the author developed an algorithm that uses a distinct, effective ROI to find skin lesions. To enhance segmentation, the authors developed adaptive region expansion. Atypical, nevus, and malignant lesions may all be classified using convolutional neural networks (CNN). The method is examined in the image processing program of MATLAB. The data shown are BCC, Dermatofibroma, LNOS, and Lichenoid Keratosis. The ratio of typical to aberrant photos is 80/20. They identified that the KNN classifier was inefficient. The accuracy and specificity of identification are increased by using a CNN classifier. Only CNN-based machine learning could compete with FFNN. In their inquiry, a complete recognition rate is contrasted. The algorithm performed well on the ISIC research dataset, and their study focused on Melanoma, Nevus, and Atypical.[11]

Skin disease classification using skin images At the 4th ICECA in 2020, Using Transfer Learning Technique was released. CNNs were used to detect skin cancer lesions. They made advantage of the available ICIS dataset. Nearly ten thousand pictures of benign diseases and melanoma are included in the data collection. Janoria, Minj, and Patre's CNN model 2016 retrieved 1,000 characteristics from a single picture. The VGG 16 architecture of the CNN model is used for feature extraction in both the input and output pictures. a helpful model that can take a certain picture and extract 1,000 attributes from it. The accuracy of the K Nearest Neighbor, Linear Discriminate Decision Tree was 99.9%. They found that ensemble learning can only increase by 48.2%. The performance of the transfer learning model is excellent for a broad range of Convolution layer classifier combinations. Both the decision tree and the K closest neighbor filters have classification accuracy levels exceeding 95%. They point out that the ROC curves of both classifiers show an almost undetectable incorrect prediction rate. Studies reveal that certain models, such as K nearest neighbors, are more than 95% accurate after several iterations. However, complex models with accuracy of less than 50%, such as supervised techniques with improved trees, perform poorly. Instead of being coarse multiclass models, classifiers are specified as nonlinear binary classifiers.[17]

K. Sriwong, S. Bunrit, K. Kerdprasop, and Nittaya Kerdprasop authored a study titled "Dermatological Classification Using Deep Learning of Skin Image and Patient Background Knowledge" in December 2019. The Suranaree University of Technology funded this study (SUT). They used deep learning techniques including Unsupervised Pretrained Networks, Recurrent Neural Networks, Recursive Neural Networks, and CNN to analyze picture data and patient background information (using three primary layers). In this article, the CNN deep learning model is utilized to automatically categorize skin disorders. They include both visual data and patient-specific information into the modeling procedure to enhance CNN's classification performance. The CNN model can diagnose skin diseases with an accuracy of 79.29%, according to experimental findings using a public dataset. However, their suggested technique of integrating patient-specific background information during the modeling phase may enhance the accuracy to 80.39%.[30]

The approach that utilizes many deep learning/machine learning models has the greatest effect on recognition. Dermoscopy is a high-resolution imaging technique (non-invasive approach for skin imaging) that enables seeing the structure of the skin where the lower epidermis and higher dermis meet. The most often used dataset for skin diseases is ISIC 2018. DensenNet, Inception, VGG, AlexNet, and ResNet are popular CNN architectures that are utilized for image recognition utilizing deep learning. The best Model technique is the multimodal fusion approach based on deep learning.

Machine learning in skin disease is assessed using the accuracy (Acc), mean average precision (MAP), false-positive rate (FPR), true positive rate (TPR), and true negative rate (TNR) metrics (TNR). In order to identify research contributions, this work will use a number of supervised learning algorithm to identify specific diseases. This article provides an overview of the performance assessment of several supervised learning algorithms used to forecast skin conditions. This important piece of knowledge may assist other researchers in deciding which supervised machine learning algorithms to utilize for their project.

Researchers in data science have recently focused a lot of their emphasis on both disease prediction and medical information. The primary goal of this research is to compare the performance of several supervised learning algorithm versions. Large-scale health datasets are readily available, and computer-based technology is widely used, which has encouraged data science researchers to focus increasingly on the Skin care industry.[29]

2.2 Categories of skin cancer

Skin cancer is the uncontrolled growth of skin cells. It typically appears on or in skin areas that are exposed to a lot of sunlight, which damages DNA and causes mutations to occur. These flaws cause the uncontrollable growth of skin cells, which leads to malignant tumors. Skin cancer falls into two primary categories: melanoma and non-melanoma. These are the most prevalent types of cancer among all those that can develop in people, with melanoma ranking 19th and non-melanoma at number five globally, respectively, according to the World Cancer Research Fund. These numbers are probably underestimates.

The 7 distinct types of skin growths and lesions that our paper will discuss include:

1. Benign keratosis -like lesions

2. Actinic keratoses and intra-epithelial carcinoma (aka Bowen's disease)
3. Basal cell carcinoma
4. Dermatofibroma
5. Vascular lesions
6. Melanoma
7. Melanocytic nevi

Benign keratosis-like lesions

Benign keratosis-like lesions, also known as seborrheic keratoses, are benign (non-cancerous) skin growths. Seborrheic keratosis is quite frequent. Middle age is where it usually starts, and as you get older, you can acquire more of it. Despite sometimes looking like other skin growths that are cancerous, seborrheic keratoses are not precancerous. Roundish or oval-shaped skin patches with a "stuck on" appearance are known as seborrheic keratoses. A waxy or scaly description has been used in the past. Even when they are flat, your finger may still feel them since they are elevated above the skin. Typically, they are brown, although they can also be black or tan, and less frequently, pink, yellow, or white. The lesions are frequently present.

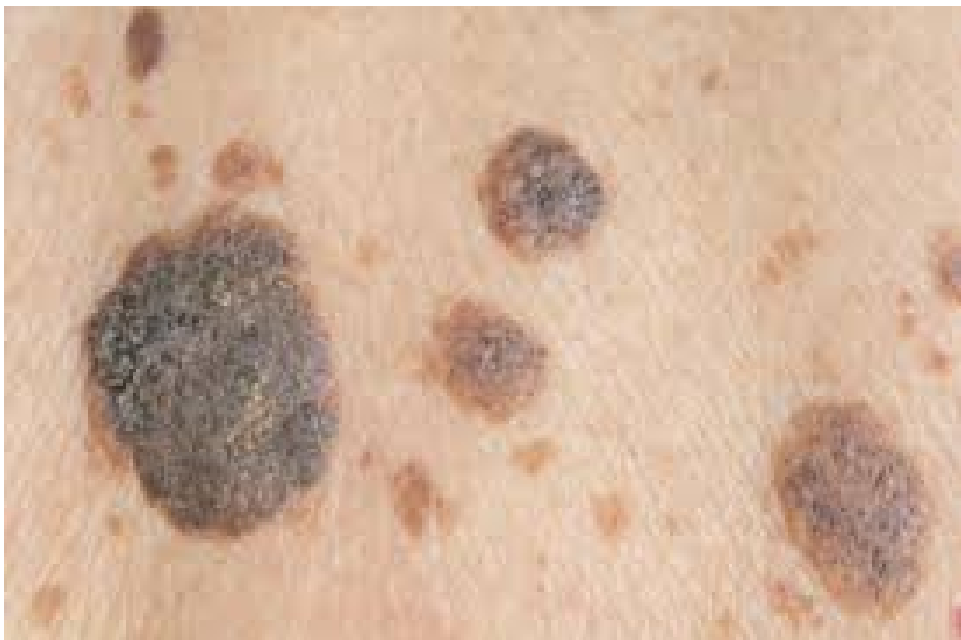


Figure1 : Benign keratosis-like lesions

Seborrheic keratoses have been observed in an estimated 90% of all adults over the age of 30, but are relatively rare in people under the age of 20 despite being extremely widespread across the whole population. Over time, the lesions grew throughout the

body.

Actinic keratosis and intraepithelial carcinoma

Actinic keratoses (AKs) are cutaneous premalignant lesions that can develop into squamous cell carcinoma. They are a typical skin ailment that appears on persons who have had a cumulative amount of sun exposure. It is also known as Bowen's Disease. These are all considered as types of malignant keratinising tumors. On parts of the body with keratinoid cell structures, such as the scalp, ears, arms, neck, and backs of the hands, it typically presents as a number of pink papules (skin lesions) with a scaly surface. Actinic keratoses often arise from the skin damage caused by ultraviolet (UV) radiation after years of sun exposure. Actinic keratoses often take one of two major forms:

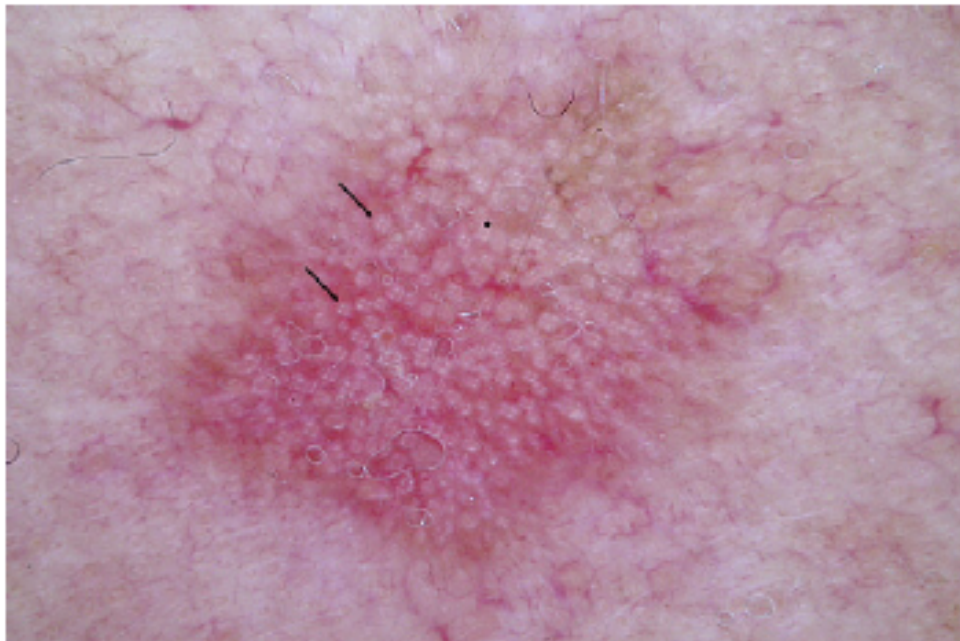


Figure2 : Actinic keratosis

Pigmented Actinic Keratosis :With reported markers such uneven follicle openings and the appearance of grey specks grouped in symmetrical patterns between follicular structures, this variant of AK is thought to primarily afflict those with darker skin. They connect all of the impacted follicular structures in a noticeable pseudo-network.

Non-Pigmented Actinic Keratosis: Non-pigmented versions of AK exhibit 4 distinct characteristics that permit a precise diagnosis with high specificity:

1. Fine, wavy vessels - Surrounding hair follicles these vessels are formed.
2. Targetoid hair follicles - holes on the hair follicles that are encircled by a white circle and have a yellowish solid in the center.
3. Erythema - Skin that is pale red and has no discernible hypopigmentation in any

shape.

Pink-to-Red pseudo-network - comparable to erythemas, but having a network-like form structure that corresponds to skin follicles.

Basal Cell Carcinoma

Basal cell carcinoma is the most frequent type of cancer that occurs on skin and the most common cancer overall. Basal cell formation that is abnormal and unregulated is what leads to BCCs. One of the three basic cell types in the epidermis, basal cells, shed when new ones form. BCC most usually occurs when DNA damage from ultraviolet radiation from the sun causes changes in basal cells in the epidermis, which is the skin's outermost layer. Unchecked growth is the result of these modifications.



Figure3 : Basal Cell Carcinoma

BCCs hardly ever spread beyond the original tumor site. But because of their propensity to spread and deteriorate, these lesions can be fatal. If BCCs are not treated, they can spread widely and deeply into the skin, leading to damage to the bone, skin, and tissue degradation. The BCC is more likely to return, often frequently, the longer you wait to get treatment. When BCC spreads to other areas of the body, there are a few extremely rare, aggressive cases. This kind of BCC can be fatal in extremely rare circumstances.

Dermatofibroma

On the skin of the lower legs, dermatofibroma, a common benign fibrous bump, is frequently seen. Small, innocuous growths that appear inside the skin, they are often pink in color on fair-skinned persons and brownish on darker-skinned people. The growth may become darker if the irritated surface is scratched. Adults typically

develop dermatofibromas. Dermatofibromas can happen to individuals of any race. Women are more likely than men to develop regular dermatofibromas, though some histologic variants are more frequently found in men. When touched, these growths have a thick texture that gives the impression that a lump or small stone has developed beneath the skin. In order to recognize dermatofibromas from other growths, they typically cause the skin around them to dimple. Even though these growths are usually painless, they might cause itching, discomfort, or irritation.



Figure4 : Dermatofibroma

Vascular Lesions Vascular lesions, often known as birthmarks, are relatively common anomalies of the skin and underlying tissues. Vascular lesions can show as an inexplicable mass devoid of specific clinical symptoms, but they can also exhibit the traditional clinical look of a superficial soft tissue swelling with a bluish color an audible bruit.



Figure5 : Vascular lesions

Vascular lesions might fall into one of two groups, each of which differs from the others in terms of where they came from or the care they require. These are the categories:

Hemangiomas: Hemangiomas and haemangiomas are vascular tumors made of blood vessel cell types that are often benign. The most frequent type of infantile hemangioma, also referred to as a "strawberry mark," is observed in children. It typically appears on the skin at birth or in the first few weeks of life. Even though hemangiomas can develop anywhere on the body, they most frequently do so on the face, scalp, chest, or back. They frequently continue to expand for up to a year before beginning to contract as the child ages. If a hemangioma impairs breathing, eyesight, or is likely to result in long-term deformity, treatment may be necessary. In a small percentage of cases, internal hemangiomas might either be the root of or exacerbate other health issues. After six to twelve months of growth, they stabilize, begin to grow smaller over the course of several years, and then eventually stop growing altogether. In some newborns, hemangiomas of the eyelid can cause irreversible blindness.

Pyogenic Granulomas: Small, swollen, and red pimples on the skin are called pyrogenic granulomas. The bumps may or may not be wet, and they have a flat surface. Because there are so many blood arteries there, they bleed readily. It is a harmless growth that is not malignant. After an injury, skin lesions known as pyrogenic granulomas may appear. A pyogenic granuloma begins as a lesion that grows quickly for a few weeks on average. After then, it settles as a raised, reddish nodule that is normally less than 2 cm in diameter. It depends on how much it bleeds whether the lesion is smooth or has a rough or crusty surface.

Vascular Malformations: Congenital defects include vascular abnormalities. Vascular anomaly that may only impact veins, only lymph vessels, both veins and lymph vessels, both arteries and veins, or neither of these. When venous malformation (VM) develops, just the veins themselves are impacted. When lymphatic dysfunction happens, only the lymph vessels are impacted (LM).

Melanoma

The cells known as melanocytes, which give skin its brown or tan color, have the potential to beyond their confines, which can result in the growth of melanoma, a particular type of skin cancer. Melanoma is a very uncommon kind of skin cancer in comparison to numerous other types. However, melanoma poses a greater threat because it has a much higher propensity to extend to other parts of the body if left undetected and untreated. Malignant melanoma and cutaneous melanoma are other names for this malignancy. The majority of melanoma cells still produce melanin, which causes melanoma tumors to frequently be brown or black in color. While some melanomas do develop melanin, others may not, giving them a pink, tan, or maybe white appearance. Melanomas can develop any place on the skin, but for women they typically start on the legs, while for men it's usually the chest and back. Other typical locations are the neck and face. The likelihood of getting melanoma at these more common sites is lower if you have darkly pigmented skin, yet anyone can get it on their hands, feet, or under their nails. The cells in our skin and eyes known as melanocytes are responsible for producing and storing the pigment melanin, which gives our outward appearance its color. Like all skin cancers, melanoma is brought on by injury to the skin. UV radiation damages DNA, which causes uncontrolled cellular growth melanocytic cells.



Figure6 :Melanoma

Melanomas come in a variety of sizes, shapes, and colors, and are frequently mistaken for melanocytic nevi. For this reason, it can occasionally be challenging to identify a specific set of symptoms for melanoma. A few methods include symmetry, color, and evolution to distinguish between melanomas and melanocytic nevi. Typically, most melanomas exhibit asymmetry and are occasionally colored with several colors. Additionally, melanomas frequently change in size, texture, and color over time or perhaps getting scratchy, or worse, bleeding

Fairer skinned individuals have a lifetime risk of melanoma of less than 3%, which is more than 20 times higher than that of darker skinned individuals. Despite the fact that risk percentage rises as the level of danger does. Melanoma remains one of the commonest malignant types of people regardless of age. There are numerous varieties of melanomas, some of which include following list:

Superficial Spreading Melanoma: This type of melanoma is the most prevalent and spreads for a number of years along with the epidermis. These frequently have uneven borders and are rough. Before entering tissue layers beneath the epidermis, it may continue to grow along the skin for a few of years.

Nodular Melanoma: A lump or node that appears above the skin's surface and is firm to the touch accounts for 15% of all melanoma occurrences. The majority of the time, they resemble moles or melanocytic nevi.

Lentigo Maligna Melanoma: About 5% of all cases of melanoma that have been documented are of this unusual form. Another name for it is "Hutchinson's melanotic freckle." This is one of the closely related forms of melanomas more than other kinds of melanomas, is connected to excessive sun exposure. This is due to the increased risk of lentigo maligna melanoma in older adults due to their prolonged exposure to the sun's UV radiation throughout their lifetime. The face is where

diagnoses are made the most frequently.

Acral-lentiginous Melanoma: The most prevalent type of melanoma to affect people with darker skin is acral-lentiginous melanoma, which affects African Americans 70% of the time and Asians 46% of the time. Melanoma on the palms of your hands, the bottoms of your feet, and the spaces between your finger and toe nails are commonly occurring growth locations.

Melanocytic nevi

The benign skin lesion known as a melanocytic nevus, or more often known as a mole, is caused by the overgrowth of melanocytes, or pigmented cells. Dark moles that can be seen on both men and women are produced by the melanocytes colliding. These can either be present at birth or become apparent later in life, and can affect people of all ages.



Figure 7 :Melanocytic nevi

Fair-skinned persons are more likely to have melanocytic nevi than people with darker skin, despite this condition commonly affecting all people. These can develop anywhere on the body, although they typically appear on the limbs. They can have a variety of colors and textures (rough, raised, flat, or thickened). They can occasionally be hard to distinguish by color, ranging from flesh tones to black through images. There are some nevi present along with excessive hair growth.

Although studies has indicated that genetic susceptibility and quantity of sun exposure are factors, the precise reason why melanocytic nevi appear is obscure. Melanocytic nevi can alter in size, shape, and color throughout time, whether they are new or existing being prone to enlargement and unform. Higher nevi counts are associated with melanoma than those with less have been shown to have a higher

risk of developing nevi. Small melanocytic nevi are present at birth in 1 in 75 individuals, but larger nevi appear in per 20,000 births.

The malignant form of melanocytes known as melanoma, which frequently results in skin cancer death, should not be confused with melanocytotic nevi. Nevi typically have symmetrical shapes, which can help differentiate them from melanoma.

Chapter 3

Dataset Description

It takes a sizable amount of sorted and annotated images to train a diagnostic algorithm built on a neural network. But sources providing trustworthy assessments of top-notch dermatoscopic images are either unreliable or have a little amount of data.

The dataset we are using here is , a large collection of annotated images has been collected into a dataset named "HAM10000 (Human Against Machine with 10000 training images)" The lack of diversity and small size of the datasets of dermatoscopic images available hinder the training of neural networks for automated diagnosis of pigmented skin lesions. Using the HAM10000 ("Human Against Machine with 10000 Training Images") dataset is how we address this issue. Dermatoscopic images from various populations that had been taken and preserved using various modalities are gathered. Several approaches for data collection are used and cleaning due to this variability, and we semi-automatic procedures are created leveraging specially trained neural networks. 10015 dermatoscopic images make up the final dataset, which is made available to the public through the ISIC archive and published as a training set for scholarly machine learning applications. When compared against human experts, this benchmark dataset can be used for machine learning. All significant diagnostic subtypes for pigmented lesions are included in the cases as a representative sample. Pathology has validated more than 50% of lesions, while the other cases relied on follow-up, expert consensus, or confirmation by in-vivo confocal microscopy as the ground truth.[6]

| | |
|--------------------------|---|
| Measurement Type(s) | Skin lesions |
| Design Type(s) | Dataset creation objective; Data integration objective |
| Sample Characteristic(s) | Homo sapiens;skin of body |
| Factor type(s) | Diagnosis:Diagnostic procedure:Age ; Biological sex;Animal body part |
| Technology Type(s) | Digital curation |

By using Ham10000 dataset images of various skin cancer are firstly converted from jpg to tiff. Then using the imageJ we annotate the images. Image annotation is specified as the task of labeling digital photos using human input and, in some circumstances, computer-assisted assistance. A machine learning engineer selects

labels which provide information to the computer vision model about the objects in the picture. Labeling is done using various annotation methods such as bounding boxes, shapes, key spots, and more. We used the pen tool to precisely outline the dark spot from the image. This allows to ensure that tap the outer edge to cover the spot, that will be highlighted with a specific color to differentiate it from the unaffected skin area. Moreover, pixels are added more to annotate more lesions and spot it. By this it was able to spot the affected or illness to get correct accuracy in segmentation. In annotation we use mask. The Generate Annotation Masks tool generates masks for annotation or other features that intersect another layer, or a list of additional layers, that you designate. They blur an image to minimize the edge content and to make the transitions between various pixel intensities as smooth as feasible. Noise reduction may also be conducted using the technique of blurring.[15]

3.1 Dataset Background

When compared to inspection with the naked eye, the diagnostic accuracy of dermatoscopy, a widely used diagnostic method, for benign and malignant pigmented skin lesions, is improved¹. To train artificial neural networks to automatically identify pigmented skin lesions, dermatoscope photos are also a good source.

The most deadly form of skin cancer, melanomas, were successfully distinguished from melanocytic nevi using dermatoscopic pictures in 1994 by Binder et al. The study, like the majority of investigations, was limited by a small sample size and a dearth of dermatoscopic images other than melanoma or nevi, even though the outcomes were positive. Because of recent progressions in graphics card capabilities and machine learning techniques that establish new standards for neural network complexity. It is expected that very soon without the help of human expertise, all types of skin lesions with pigment can be treated using automated diagnostic systems.

Although there are few or just a few types of disorders represented in high-quality dermatoscopic images with reliable diagnoses, training neural-network based diagnosis algorithms necessitates a huge number of annotated images.

160 nevi and 40 melanomas were included in the 200 dermatoscopic pictures that Mendonça et al. released as the PH2 dataset in 2013. For melanomas, pathology provided the fundamental information, but most nevi did not. The collection served as a benchmark dataset for research of the computer diagnosis of melanoma up until this point because it is openly accessible and contains extensive metadata.

A CD-ROM comprising digital versions of 1044 dermatoscopic images, including 167 images of non-melanocytic lesions and 20 photos of diagnoses not included in the HAM10000 dataset, is commercially available to go along with the book *Interactive Atlas of Dermoscopy*. Despite being one of the databases with the widest range of covered diagnoses, its use is definitely hampered due to its difficult accessibility.

As of April 2020, there were roughly 23,665 images in the ISIC archive, a freely accessible collection of dermatoscopic images. Melanocytic lesions and nevi make

up around 90% of the dataset, which is strongly specialized in this area. Because of its vast size, organized structure, and licensing options, it is currently the industry standard for finding dermatoscopic images for research purposes.

The distinction between melanoma and nevus was made in previous study by focusing on melanocytic lesions because to the constraints of the datasets available, whereas non-melanocytic pigmented lesions—which are more frequent in practice—were ignored. A mediocre performance of automated diagnostic systems in the clinical context was achieved despite outstanding performance in experimental settings due to the mismatch between the tiny diversity of training data and the variety of real-world data. Contrary to binary classification, creating a classifier for several diseases is more difficult. At this time, dermatoscopic images cannot be reliably predicted using multi-class methods for clinical imaging of skin diseases.[10]

| Abbreviation | Class | No. of Images |
|--------------|-------------------------------|---------------|
| akiec | Bowen disease | 334 |
| bcc | Basal Cell Carcinoma | 583 |
| bkl | Benign Keratosis-like lesions | 1674 |
| df | Dermatofibroma | 122 |
| mel | Melanoma | 2177 |
| nv | Melanocytic Nevi | 18,618 |
| vasc | Vascular Lesions | 157 |
| | Total=23665 | |

Table 3.1: Overview of ISIC archive and Class Distribution upto April 2020

3.2 Classification of Dataset

As seen in the table above, the photos in the HAM 10000 dataset were gathered over a 20-year period from the Department of Dermatology at the Medical University of Vienna in Austria and Cliff Rosendahl’s skin cancer clinic in Queensland, Australia. Prior to the invention of digital cameras, the Medical University of Vienna started gathering its data and photos which resulted in photographs and data information being saved in various formats depending on the time of capture. On the other hand, the Australian method stored its metadata in Excel databases and its photographs in PowerPoint files. In order to maintain consistency and make the data accessible, it was necessary to sort and categorize the data into a single form of formatting. Among the techniques was

Extraction of images and metadata from PowerPoint files: One PowerPoint file was created by the Australian skin cancer practice for each calendar month’s worth of diagnostics, which included their monthly clinical reports and photographs. A single image, a remark, and an individual identification number were all present on each slide of the monthly files. Tschandl et al. mechanized the process of iterating through all of the files and categorizing the content based on the identifiers

assigned to each one using Python.

Diapositives are digitalized: Prior to the development of digital cameras, dermatoscopic images were kept in the Department of Dermatology in Vienna as diapositives. These diapositives were digitized, then saved as 300 DPI, 8-bit JPEGs. Following a manual histogram to improve the color and contrast of the scan in comparison to the original, images were cropped with the lesions placed in the middle of them at a 800x600px ratio.

Digital dermatoscopy System Data Extraction: The Department of Dermatology switched to using the MoleMax HD, a digital dermatoscopy device, years after digital cameras were first introduced. The system's data was extracted using proprietary software. Benign non-melanocytic lesions, melanocytic nevi with more than 1.5 years of digital dermatoscopic follow-up, and excised lesions with a histopathologic report were the only three categories of data that were gathered (which were later manually matched to their respective lesions). In order to center the growths or lesions in the photograph, these images were manually cropped to a 800x600px resolution.

Dermatoscopic image filtering: The medical staff took all of the pictures, which included both dermatoscopic and clinical close-ups. The dermatoscopic images had to be manually isolated from the other photos because these additional images lacked reliable source annotation. First, 1501 photos from the Australian data set were manually categorized into three categories: overviews, close-ups, and dermatoscopy. The remaining images were then automatically classified using an ANN that had been trained using these manually labeled images. The HAM dataset's picture categorization procedure may now be completed more quickly because to the top-1 accuracy of 98.68% that was attained. This is followed by a second manual inspection to correct any misclassifications.

Pathological diagnosis that are consistent: Recorded biopsies or histopathological diagnose Typos, unsure diagnoses, various terminologies and classifications, and many different diagnoses per lesion were among the numerous discrepancies found in the data from both the Australian practice and the Department of Dermatology.

Except for melanomas that were connected to nevi, uncertain diagnoses were disregarded.

Except for melanomas that were connected to nevi, uncertain diagnoses were disregarded. In order to eliminate any ambiguity in generalization, the remaining diagnoses were divided into seven (7) generic classifications. More than 95% of all pigmented lesions and growths from both data retrieval sites were covered by these groups, which are further addressed below.

A final manual review and validation of every image in the data collection was performed to weed out any information that might have contained

- Identifiable material (tattoos, garments, etc.)

- Images that are hazy or out of focus
- Images with artifacts (visible anomalies)
- lesions without pigment
- Overviews and close-ups that the ANNs did not immediately eliminate

3.3 Technical Validation

In order to manually correct histograms for human viewers' ease of visual access, specialist dermatologists were engaged as necessary. The only shots that required correction were those that were underexposed and those with a noticeable tint that distorted the subject's natural skin tone. The grey-world assumption, a method of determining the white-balance of an image that assumes that the majority of the image is a neutral gray color, was used to test illuminant color estimations after rectification. In order to compare the photographs before and after the histogram adjustments, it is possible to graph the average color contained in the image, as seen in the image below. This demonstrates the improvement in image clarity.

3.4 Technical Terms and Definitions

For convenience of use and data organization, every image in the HAM10000 dataset has been categorized into a metadata csv file according to seven classification categories, each of which has many sub-categories. Which are:

Lesion Identification (lesion_id, image_id)

The "lesion id" in the metadata file that identifies each distinct lesion recorded in the HAM10000 dataset is written next to every image in the dataset. The prefix "HAM_" comes after these IDs. Additionally, each image has a special identification number, or "image id," that links it to the relevant image file in the ISIC database and is prefixed with "ISIC_."

Additionally, the dataset contains data for pixelated, flattened 28x28x3 versions of the images that are available (HMNIST 28 28 RGB), where each image is connected with a label between zero (0) and six (6). (According to the metadata file, each label relates to the diagnosis (dx) in the preceding order (1 is akeic, 2 is bcc, etc.)

Lesion Diagnosis (dx, dx_type)

According to their particular medical diagnoses, each image in the dataset is categorized using seven (7) distinctive identifying categories under the category type of "dx."

bcc- Basal Cell Carcinoma

bkl- Benign keratosis-like lesions

df- Dermatofibroma

Akiec- Actinic Keratoses and Intraepithelial Carcinoma (aka Bowen's Disease)

vasc- Vascular Skin Lesions

mel- Melanoma

nv- Melanocytic Nevi

Chapter 4

Pre-processing of Dataset

The rate of success of machine learning may be influenced by a variety of factors. The first priority is to have accurate data and a proper representation. If the dataset contains duplicate data, the algorithm's training phase will end up being more challenging and complex. Due to its capabilities for feature extraction, transformation, data cleaning, and normalization, data preprocessing is a great solution to this kind of issue. This method produces the final training set of data. For The data required for the prediction process may be extracted from the data preparation stage.[21]

The training of neural networks for automated detection of pigmented skin lesions is hindered by the collection's limited size and lack of variation. To tackle this issue using the HAM10000 ("Human Against Machine with 10000 training images") dataset. Here, by gathered dermatoscopic pictures captured and archived using a number of modalities from persons of various demographics. The final dataset, which might be utilized as a training set for research-based machine learning, has about 10015 dermatoscopic pictures. Actinic keratoses and intraepithelial carcinoma, Basal cell carcinoma, benign keratosis-like lesions, dermatofibroma, melanoma, melanocytic nevi, and vascular lesions are also included in the instances.

More than 50% of lesions are diagnosed by histology, whereas the other instances are confirmed by follow-up testing, expert consensus, and in-vivo confocal microscopy. Lesions in the dataset with several pictures may be tracked using the lesion id column in the HAM10000 metadata file.

The first stage in the detection process is pre-processing to enhance the picture quality for subsequent usage by eliminating undesired image information, often known as image noise, which is present in raw photos. If this issue is not done properly, a variety of classification errors may occur. The need for this pre-processing, in addition to the imprecision, is brought on by the low contrast between the skin lesions and the surrounding healthy skin, as well as aberrant boundaries and skin deformities, such as hair, shadow, skin lines, air bubbles, reflection, and black frames. A number of filters, including the median filter, the adaptive median filter, gaussian filter and the adaptive Wiener filter, may be used to remove Gaussian noise. For instance, misdiagnosis may occur if a picture of a lesion included hair. The pre-processing steps of adjusting contrast, erasing vignetting, color-correcting, smoothing the picture, eliminating hair, normalizing, and localizing are expected to eliminate or fine-tune

image noise. The right mix increases the accuracy of pre-processing activities, In order to increase the precision of segmenting skin lesion pictures, we employed to enhance the training data set. It is possible to use effective techniques that rely on data augmentation by rotating the picture and applying horizontal flipping resizing, lighting correction, hair removal, and contrast enhancement. To improve the accuracy of segmentation, it has been recommended to use preprocessing. Both micrographs and dermatoscopy images require pre-processing. For the dermatoscopy image, the input for the network is RGBH (red, green, blue, and tone levels respectively), and the output is a binary segment image for red and white pixels that perform the influenced skin and non-influenced area, respectively. By employing dermoscopy pictures, we explore using SVM to categorize skin lesions. Moreover, pixels are added more to anotate more lesions and spot it. Preprocessing is a technique that removes noise from the supplied color picture. The adaptive segmentation technique is then used to separate skin lesions from dermoscopy pictures in the next step. The segmentation approach recommended provides more relevant segmentation results. However, every pictures of lessions and diseases were segmented and annotted step by step and one after another to get the exact accuracy.

Following pre-processing and picture segmentation, there is a waiting period before post-processing, when the goal is to elicit the features. Open-and-close procedures, boundary extension, region merging, island elimination, and smoothing are some of the most popular post-processing techniques used to accomplish this. Principle component analysis (PCA), textural data that is readily accessible, Gaussian derivative kernels, resolution boundary features, and area of interest are some of the approaches used for feature extraction (ROI). The ABCD (asymmetry, border, color, and diameter) basis is used to post-segment dermoscopy pictures and extract a number of characteristics.

4.1 Libraries Required

Various libraries are needed to be downloaded to make the CNN and U-net model work. The required libraries are:

- Keras
- cv2
- PIL
- matplotlib
- keras_unet_collection
- tensorflow.keras.optimizers

4.2 Cleaning of Data

Datasets that are large in size contain incomplete or null values. These need to be filled, thus we looked for the mean of these values and filled in the blanks. Due to the possibility that duplicate pictures could skew the outcomes and provide unintended results, so we have screened for duplicates and used only those images that are distinct.

4.3 Exploratory Data Analysis

Detailed information in our used dataset include the localization and diagnosis of skin growths. Moreover it includes details about the age and gender of those affected.

Distribution of Type of Cancer

Figure 8 displays the distribution of photos over seven distinct lesions groups. The displayed figure demonstrates that melanocytic nevi have the most pictures.

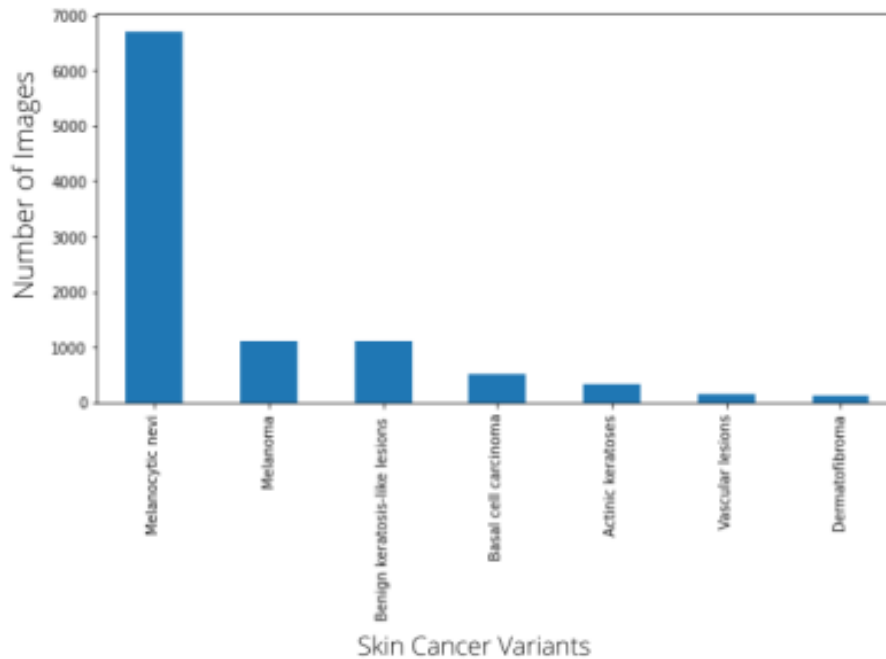


Figure 8 : Distribution of types of skin cancer

Distribution with Diagnosis Type

The dispersion of several sorts of diagnostic methods used to locate the skin growths. Figure 9 shows that histopathy contains the highest number of images which is 5000

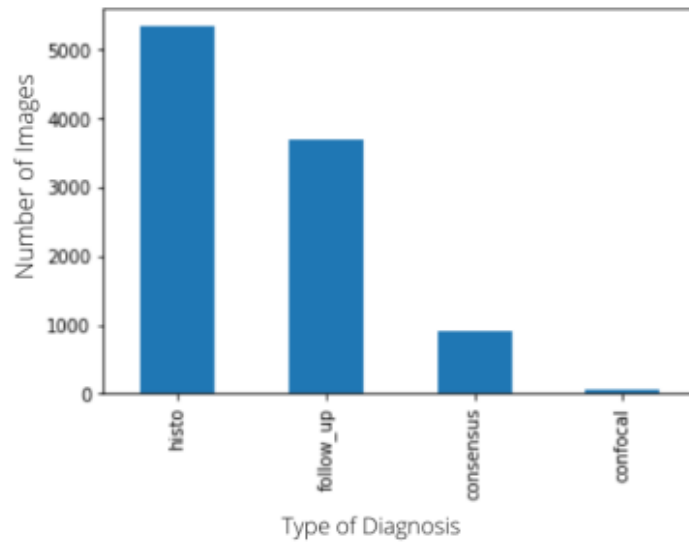


Figure9 : Diagnosis Type

Distribution with Localization

Figure 10 shows the location of the human body where skin nodules arise and how often they are may be seen from the graph below.

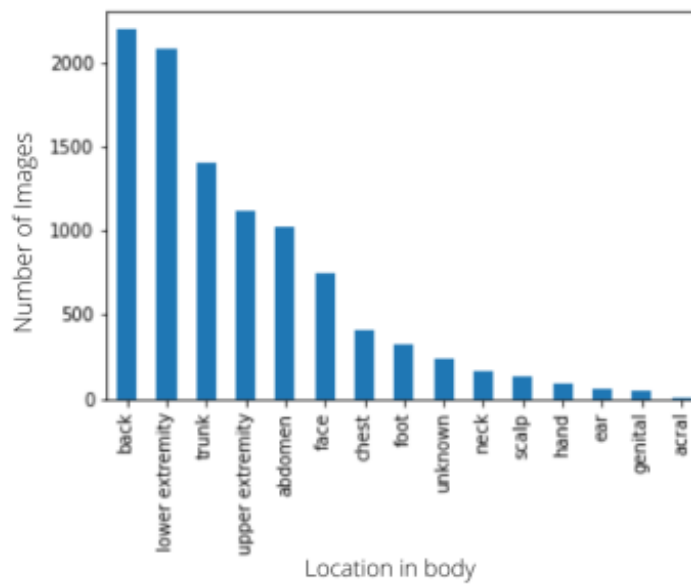


Figure10 : Localization

Distribution with gender

Figure 11 displays three types of gender - male, female and unknown and it is seen that compared to female, the percentage of getting affected by cancer for male is more.

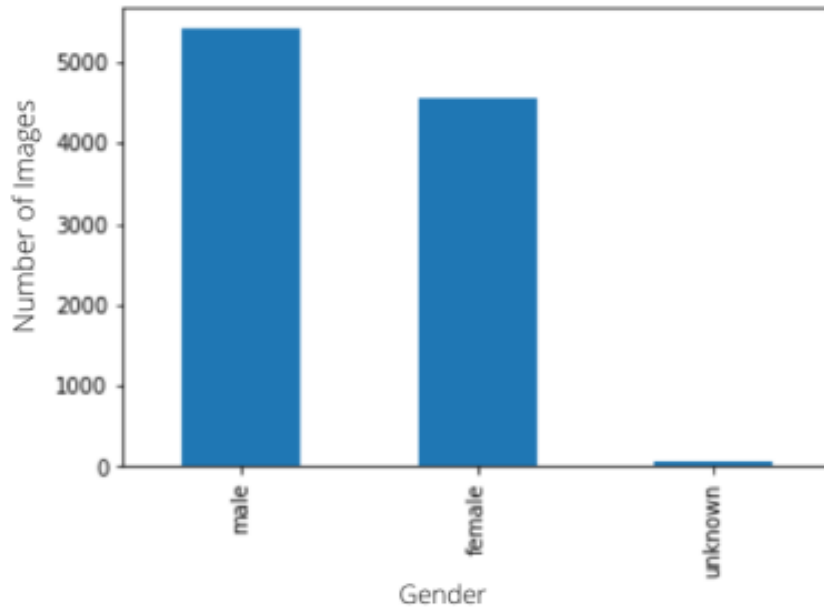


Figure11 :Gender Distribution

Distribution with age

The majority of the image data came from individuals between the ages of 40 and 70. This age group has almost as many affected individuals as all other age groups altogether.

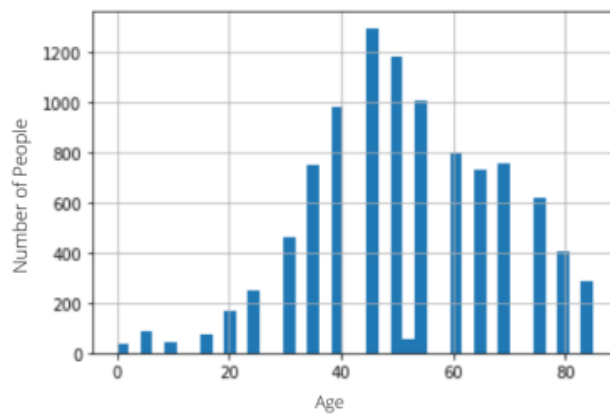


Figure12 :Age Distribution

Chapter 5

Methodology

5.1 Implementation of CNN

When it comes to our system there are 3 layers. The first layer can be taken as the input layer and in this layer is where the training of dataset takes place. The input layer's task is to gather data which are delivering and adding some weight with it which then goes to the hidden layers. The job of the neuron which are present in the hidden layers is to derive a pattern from the data[25] by separating it from its features. This is pattern is then considered as the basis of the output layer whose job is to assign to the appropriate classes and lastly to correctly select class 1 or class 0 binary classification is being used. For this case we are using class 0 to indicate the absence of harmful cells and class 1 to indicate the presence of harmful/cancerous cells.[2]

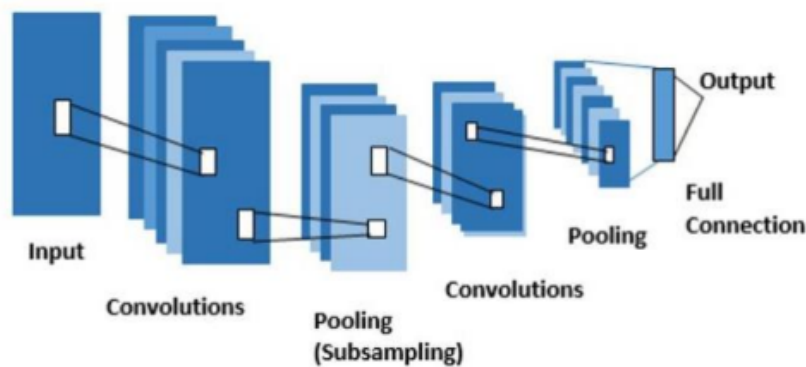


Figure 13 : multiple layers of CNN

For our method to work, we must first label our images and then we have to insert them in our dataset. The images have been organized in to 7 classes. They are defined below:

- Melanocytic nevi (nv)
- Melanoma (mel)

- Benign keratosis-like lesions (bkl)
- Basal cell carcinoma (bcc)
- Actinic keratoses (akiec)
- Vascular lesions (vas)
- Dermatofibroma (df)

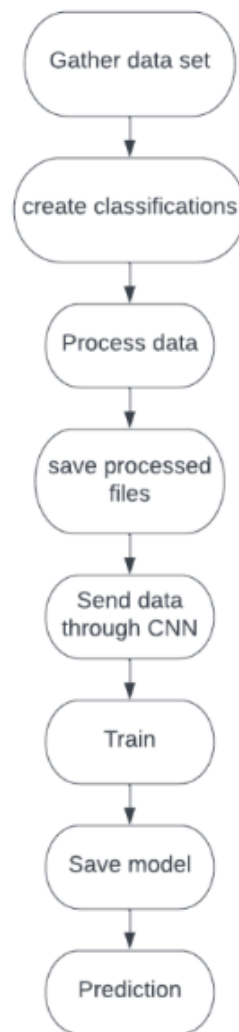


Figure 14 : CNN'S used flowchart

Steps of the Process Step 1: First step would be to gather a data set.

Step 2: In this dataset we have labelled the images which is the next step.

Step 3: After this we need to initialize every images along with all the necessary parameters.

Step 4: For the next step, a training image as input is taken by the system. It then saves the image into the system.

Step 5: Then in the next step the system predicts using CNN.

Step 6: Following this we use the CNN that was generated just before and train using that.

Step 7: Next, we save the model so that we can predict the test data.

Step 8: Finally, we evaluate the result.

Step 1(Data set): For this project we have chosen the HAM10000 dataset. This dataset contains over 10000 images which we can use.

Step 2(Label images): Next, we have to label the images. Because if the images are not properly labelled then they won't be fed to the CNN. So labelling is very important.

Step 3(Pre-processing data): A very big obstacle is the size of the images used in the dataset. This input data sometimes can be exceptionally big. The span of the picture is also another issue. The data set, which is containing the pictures can be large in width and height. To tackle these two problems when we read the images, it will be defined in such a way that only one colour channel remains. For our cases, grey scale images are generated from original images that is easier for CPU to process.

Step 4(Saving pre-processed file): The pre-processed images will be saved in the record. They will also be saved along with their classes. Classless images must be discarded as in the next step it will be fed to the CNN.

Step 5(feeding processed data to CNN): As previously mentioned, there are 3 types of layers present in CNN

- Layer of Convolution
- Layer of Pooling
- Layer of Full connection

Layer of Convolution: It can be stated as convolving an input of $x * x$ with a $y * y$ filter will result in $(x - y + 1) * (x - y + 1)$:

- Considering the input as $x * x$
- Considering the filter size as $y * y$

- Considering the output as $(x - y + 1) * (x - y + 1)$

| | | | | | |
|---|---|---|---|---|---|
| 5 | 3 | 2 | 1 | 7 | 4 |
| 3 | 5 | 8 | 9 | 1 | 3 |
| 2 | 5 | 6 | 0 | 1 | 4 |
| 1 | 6 | 7 | 1 | 0 | 2 |
| 6 | 2 | 4 | 0 | 8 | 2 |
| 2 | 5 | 4 | 2 | 3 | 9 |

*

| | | |
|---|---|----|
| 1 | 0 | -1 |
| 1 | 0 | -1 |
| 1 | 0 | -1 |

Figure 15 : a 6 by 6 image with a 3 by 3 filter

One major issue is that convolution operation means shrinking the size of the image. This may lead to information loss. So, to tackle this, padding the corners of the image or by adding borders we can minimize that.

- Considering the Input as $x * x$
- Considering the padding as p
- Considering the filter size as $y * y$
- Considering the Output as $(x + 2p - y + 1) * (x + 2p - y + 1)$

Layer of Pooling: For reducing the size of the image and increasing the speed of computing, pooling layers are used. Let us demonstrate by using a 4 by 4 matrix below:

| | | | |
|------------|-----------|-----------|------------|
| -6 | 3 | 7 | -1 |
| -15 | 6 | 19 | 1 |
| -8 | 12 | 8 | -7 |
| -6 | 10 | 4 | -10 |

Figure 16 : images for pooling layer

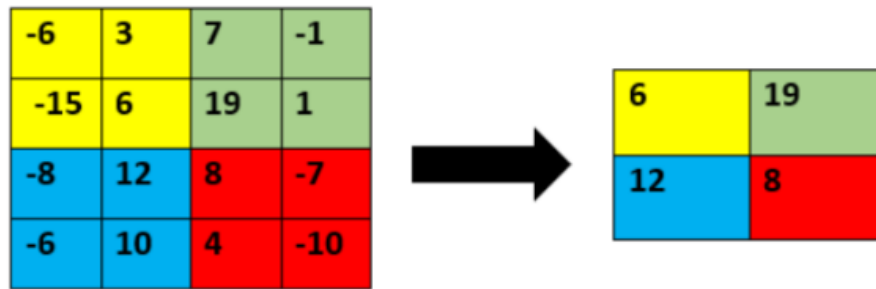


Figure 17 : Result of Max pooling

If we consider the input pooling layer to be $xh * xw * xc$, then the expected output will be $[(xh - y)/z + 1 * (xw - y)/z + 1 * xc]$

Step 6(Train): Our model needs to be trained up to 200 times. Each time the loss will decrease to a certain level. If after running the training for a certain time the changes are insignificant then we will stop the training.

Step 7(saving model): We will need the model for further testing purposes. Therefore, the model will be saved. The model will be used to predict the images. The said images might contain malignant or benign images.

Step 8(Prediction): The final output layer will be used to predict the images. After the prediction, our system will be evaluated with utmost accuracy and precision along with recall and f1 score measures.

5.2 Implementation of Unet

U-net is a deep learning architecture mostly used for images that are busy , meaning it has a lot of objects in it. For medical image detection in most cases U-net provides a better accuracy as it does semantic segmentation, semantic segmentation basically uses labels, which means it puts a label on every pixel in an image. It is used to identify a group of pixels which form a distinctive category. U-net has an architecture which forms a U shape, it has several layers which form convolutional networks and none of the layers are dense layers which means we can use images of any pixel dimensions. The architecture has a encoder at the starting and then a decoder towards the end, the first layer has a 64 filter and as we start going down the layers the filters increases from 64 to 128 to 256 and the filters start to decrease in height but becomes wider, the pixels also starts to shrink and at the bottom of the U the image is completely broken down and encoded into one tensor and then as it starts going up again on the other side the the image starts getting reconstructed and the pixels are in groups based on their labels so when it reaches the top again on the other side of the U we get a fully reconstructed image which where the pixels are in a collection based on their labels. While the image is going up the layers on the right side the information on the left side is being concatenated to the layer directly opposite to it on the right. To use this U-net architecture we needed two sets of

images, the first set composed of the greyscale images and another set composed of the masks of the images.

```

Main_multy.py x simple_unet_model.py x
from keras.models import Model
from keras.layers import Input, Conv2D, MaxPooling2D, UpSampling2D, concatenate, Conv2DTranspose, BatchNormalization, Dropout, Lambda

def simple_unet_model(IMG_HEIGHT, IMG_WIDTH, IMG_CHANNELS):
    inputs = Input((IMG_HEIGHT, IMG_WIDTH, IMG_CHANNELS))
    s = inputs

    #Contraction path
    c1 = Conv2D(16, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(s)
    c1 = Dropout(0.1)(c1)
    c1 = Conv2D(16, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(c1)
    p1 = MaxPooling2D((2, 2))(c1)

    c2 = Conv2D(32, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(p1)
    c2 = Dropout(0.1)(c2)
    c2 = Conv2D(32, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(c2)
    p2 = MaxPooling2D((2, 2))(c2)

    c3 = Conv2D(64, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(p2)
    c3 = Dropout(0.2)(c3)
    c3 = Conv2D(64, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(c3)
    p3 = MaxPooling2D((2, 2))(c3)

    c4 = Conv2D(128, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(p3)
    c4 = Dropout(0.2)(c4)
    c4 = Conv2D(128, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(c4)
    p4 = MaxPooling2D(pool_size=(2, 2))(c4)

    c5 = Conv2D(256, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(p4)
    c5 = Dropout(0.3)(c5)
    c5 = Conv2D(256, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(c5)

    #Expansive path
    u6 = Conv2DTranspose(128, (2, 2), strides=(2, 2), padding='same')(c5)
    u6 = concatenate([u6, c4])
    c6 = Conv2D(128, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(u6)
    c6 = Dropout(0.2)(c6)
    c6 = Conv2D(128, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(c6)

    u7 = Conv2DTranspose(64, (2, 2), strides=(2, 2), padding='same')(c6)
    u7 = concatenate([u7, c3])
    c7 = Conv2D(64, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(u7)
    c7 = Dropout(0.2)(c7)
    c7 = Conv2D(64, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(c7)

    u8 = Conv2DTranspose(32, (2, 2), strides=(2, 2), padding='same')(c7)
    u8 = concatenate([u8, c2])
    c8 = Conv2D(32, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(u8)
    c8 = Dropout(0.1)(c8)
    c8 = Conv2D(32, (3, 3), activation='relu', kernel_initializer='he_normal', padding='same')(c8)

```

Figure 18 : Simple Unet Model

So at first instead of having the model in our main program we created our model in a separate file as a function so that we can simply import it. This model consists of a function with the image height, width and channels and a method at the end to return the models. The algorithm here is used to define the standard architecture which has a contraction path going down and an expansive path going up. The layers declared as c show the layers going down and the layers named u are the layers going up and the information in the layers c and u are concatenated along the way.

Then comes the training function , it takes three parameters, image height, width and channel, then import the model. For training we set a batch size and an epoch,the batch size refers to the image samples which before the model is updated are processed and epoch refers to the number of complete iterations through the entire training dataset. Batch size has to be greater than 1 or equal to 1 and less than the number of samples in the dataset or equal to it. After the the training is done, check the accuracy but for semantic segmentation accuracy is not the best metric, so we use the IOU which is intersection over union where it matches the numbers to predict a possibility at every pixel, for prediction we set a threshold of 0.5 which means anything below 0.5, will be 0 meaning the black and anything over 0.5 will be 1 meaning white. Then by looking at the predicted , tested and original values the IOU score is generated. The higher the IOU score it means the pixels are getting labeled properly. After we get a satisfying IOU score then just input

external images and the run code and the system will be able to predict the affected areas of the skin from the image based on its training. It will generate some graphs and plots, the Testing Image, Testing Label and Predicted Image, and from those visual representations we are able to see how well the model is performing.

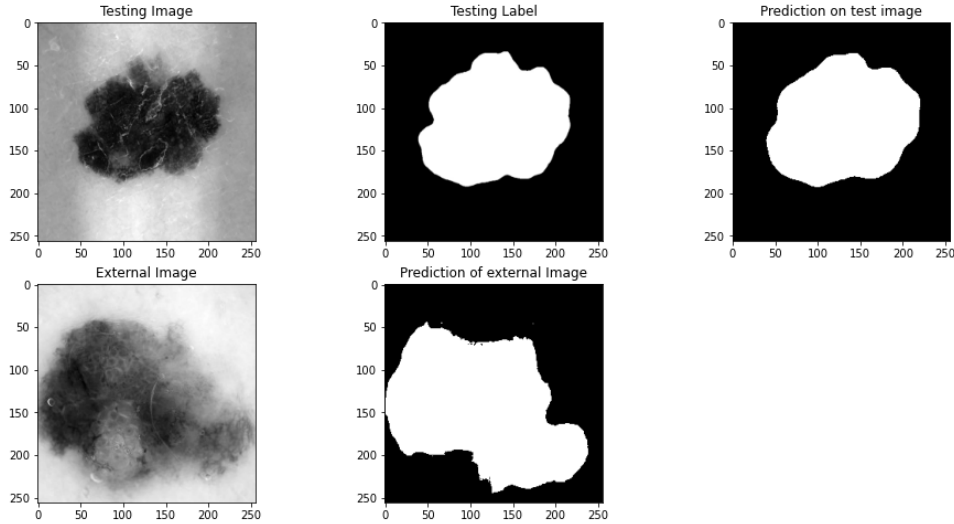


Figure 19 : Testing Image, Testing Label and Predicted Image

5.3 Implementation of Attention U-net

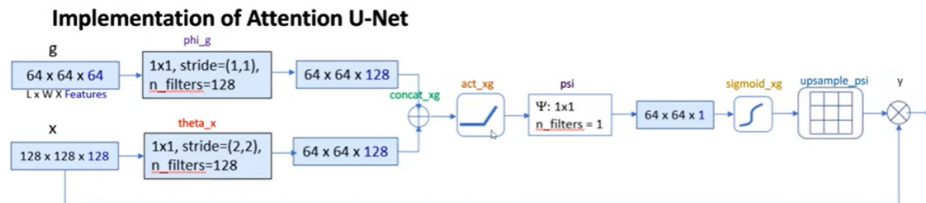


Figure 20 : Implementation of Attention U-net

So, we have our input gates g and x , g has a feature of $64 \times 64 \times 64$ and a convolution of 1×1 , $\text{stride}=(1,1)$ and no.of filters= $128 \times$ has a feature of $64 \times 64 \times 64$ and a convolution of 1×1 , $\text{stride}=(2,2)$ and no.of filters= 128 , here by putting $\text{stride}=(2,2)$ the dimensions of x goes down by half. The output we get from here for both are $64 \times 64 \times 128$. Since both of these are of the same dimensions they are added, after adding them aligned weights get larger while the unaligned weights gets smaller. After this the summation is sent through relu activation where the values below 0 is 0 and above 0 is considered as the value which has been set as the linear value. **13** After this there is a ψ with a 1×1 operation and no.of filters is 1, this in turn gives a vector or tensor of $64 \times 64 \times 1$ which is basically the weights, a sigmoid function is used to scale all the weights to between 0 and 1. After this upsampling is done using a resampler which takes it back to its original size of x which is 128×128 and once it becomes the same size element-wise multiplication is done between the output from the resampler to the original x -vector, each pixel value from x multiplied with a calculated weight value from this process this causes the weights to be updated

too as the training happens. The multiplication result allows the vector to be scaled based on relevance and then the output is passed onto the next layer.[27]

```
def attentio_block(x,gating,inter_shape):
    shape_x=K.int_shape(x)
    shape_g=K.int_shape(gating)

    theta_x=layers.Conv2D(inter_shape,(1,1),strides=(2,2),padding='same')(x)
    shape_theta_x=K.int_shape(theta_x)

    phi_g=layers.Conv2D(inter_shape,(1,1),padding='same')(gating)

    concat_xg=layers.add([phi_g,theta_x])
    act_xg=layers.Activation('relu')(concat_xg)
    psi=layers.Conv2D(1,(1,1),padding='same')(act_xg)
    sigmoid_xg=layers.Activation('sigmoid')(psi)
    shape_sigmoid=K.int_shape(sigmoid_xg)

    upsample_psi=layers.UpSampling2D(size=(shape_x[1]/shape_sigmoid[1],shape_x[2]
//shape_sigmoid[2]))(sigmoid_xg)

    y=layers.multiply([upsample,pi,x])
    result=layers.Conv2D(shape_x[3],(1,1),padding='same')(y)
    result_bn=layers.BatchNormalization()(result)

    return result_bn
```

Figure 21 : Pseudocode of Attention Block

5.4 Implementation of ResUnet

The RESUNET is made up of a bridge connecting both the encoding and decoding networks, much as a U-Net. The U-Net uses two 3 x 3 convolutions, with a ReLU activation function coming after each. In the case of RESUNET, a pre-activated residual block takes the role of these layers.

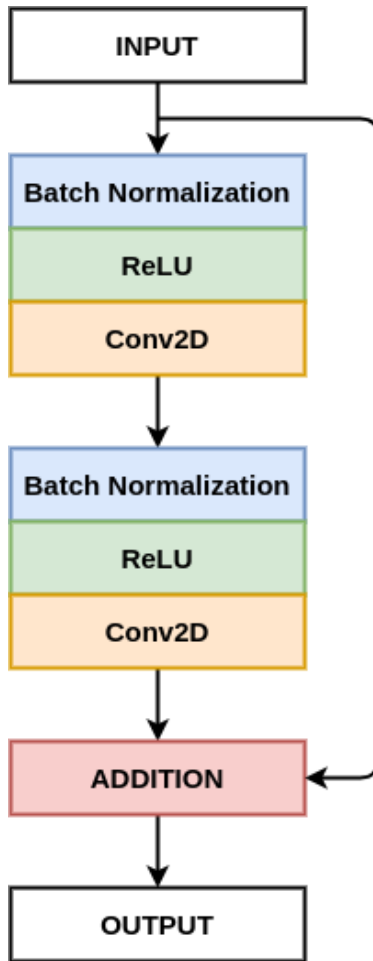


Figure 22 : Pre-activated Residual Block

The use of residual blocks makes it possible to build a deeper network without having to be concerned about gradients fading or inflating. Also it helps with basic network training. The RESUNET's many skip connections aid to improve information flow across different layers, which improves gradient flow during training (backpropagation).[14]

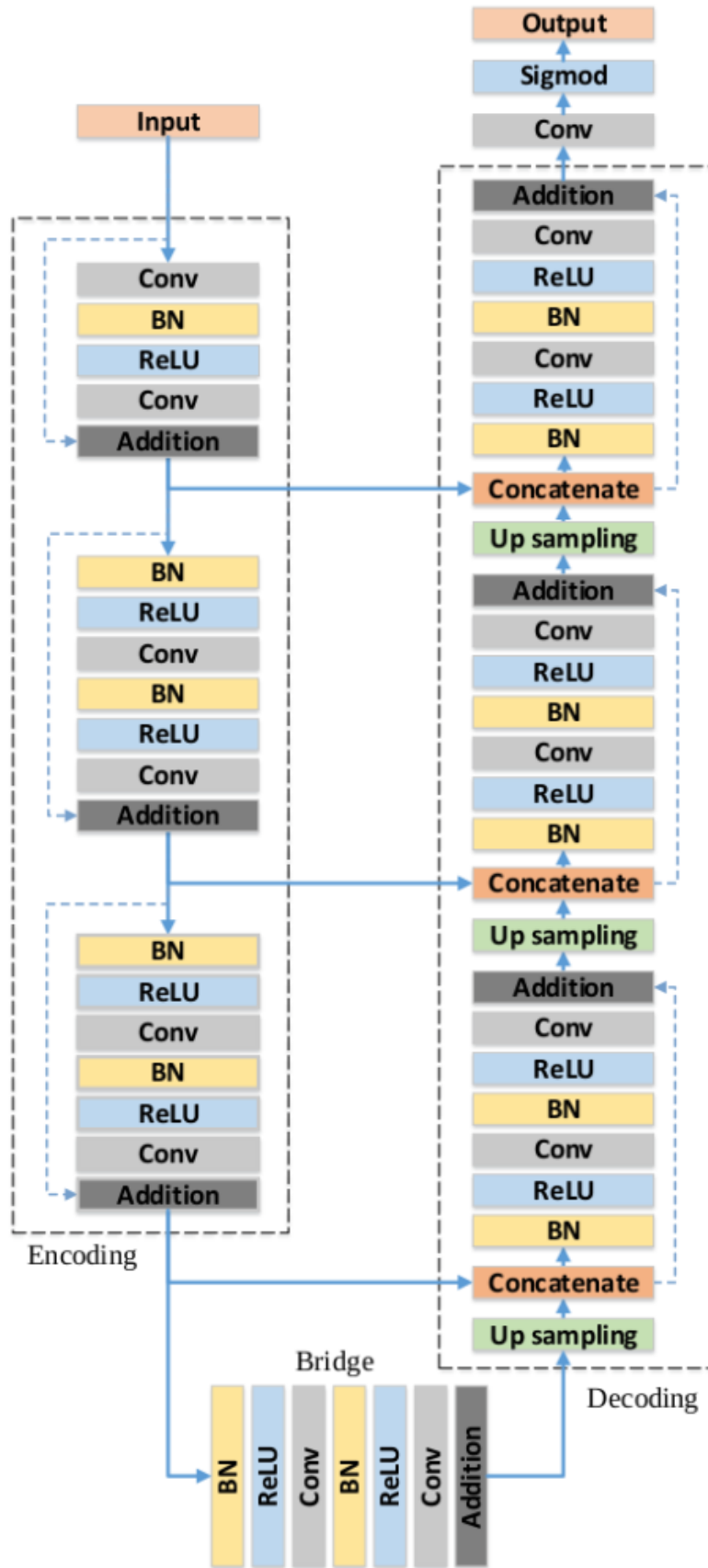


Figure 23 : Architecture of Resnet

5.5 Implementation of R2U-net

R2U-net is based on the previous model that was stated in our report which was res-unet. In a normal convolution block, input comes in and there is an output and you are mapping your output to your input via function x ($f(x)$). In a recurrent neuron, output goes back to input. There is a feed back connection. This feedback connection can happen multiple times. It depends on the value of n . After n times, it hands over the data to the next block. This recurrent network uses this feedback connection to store the information over time. They use context information as time steps increase.[13]

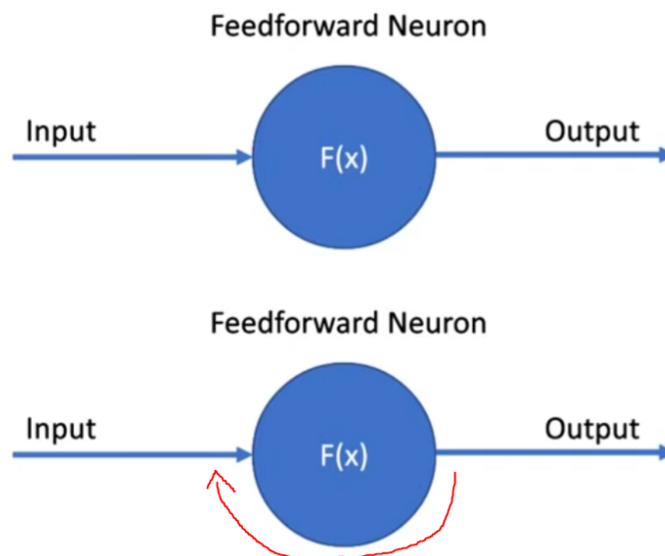


Figure 24: Basic Principle of unet

In the below image, let's take a look at a recurrent convolution block. When we unwrap one block, we can see that at times=0 it does a convolution operation. Then again at times=1 it does another one and it continues up to the max value of n . We then use this type of convolution block and create a network. That whole network is what we call a R2U-net

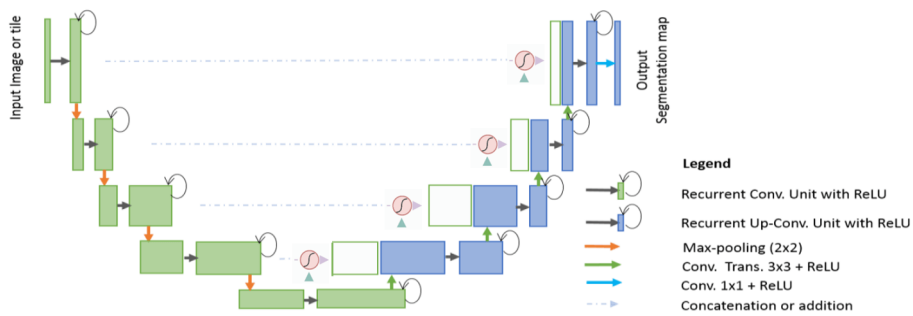
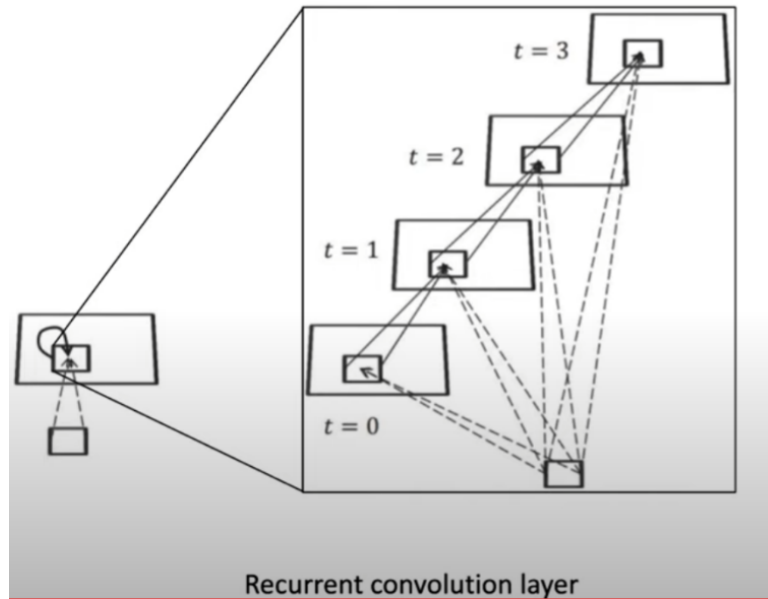


Figure 25 : Recurrent convolutional layer

UNET is the base of R2UNET. The difference is that there are recurrent convolutions. They happen before downsampling, upsampling and outputting the segmentation map.

Chapter 6

Model Building

6.1 Introduction to CNN

In order to recognize features in pixel images with the least amount of pre-processing, convolutional neural networks (CNNs) are constructed. Convolutional Neural Networks, a form of neural network with a grid-like topology, are utilized for data analysis. Digital images are essentially a binary representation of data comprised of a series of pixels integrated in a grid-like arrangement, where every cell holds a pixel value that indicates the color and brightness of each pixel.

In order to process this enormous quantity of data when we humans examine a picture, our brain needs neurons. They work together as a network and are configured to cover their whole field of vision. Each neuron in a CNN only reacts to inputs within its own receptive region. Before moving on to more complicated shapes like objects and facial features, layers are arranged in a way that splits the image down into smaller, easier forms like lines, curves, and so forth. CNN might be utilized to provide computer vision in certain aspects.

[19] Convolution, pooling, and fully connected are the three layers that typically make up a CNN. The convolution layer is where most of the computations are carried out. The matrices employed in this calculation are the kernel and the restricted region of the receptive field. The kernel matrix is more information dense than a picture, although being smaller in size. The kernel will therefore have a smaller height and breadth in terms of spatial dimensions if the image comprises three RGB channels, but the depth will cover all of them. In this layer, the two matrices' dot product is computed.

The kernel traverses the height and width of the image in accordance with the stride. The image that emerges as a result of this process is represented in 2D by the activation map. The activation map displays the kernel's response at each position in the image. The following is the formula for Output Weight during[12]

$$W_{out} = W - F + 2P/S + 1$$

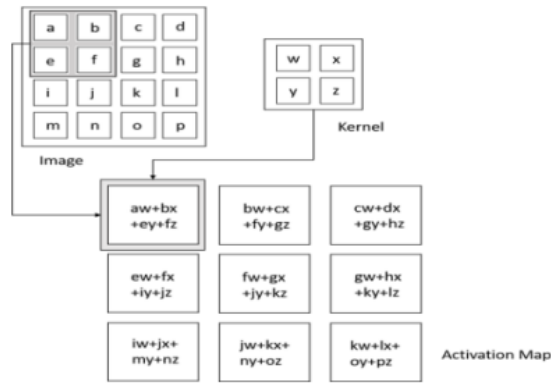


Figure 26: Convolution Operation (Source: Deep Learning by I. Goodfellow, Y. Bengio)

After that, the pooling layer adds related outputs to decide which output is highest where it is needed. As a result, computation time and weights are decreased as well as the length of the representation. We selected max pooling in our thesis because it is the most popular pooling function. Max pooling determines a neighborhood's maximum output. If we have an activation map of size $W * W * D$, a pooling kernel of spatial size F , and stride S , we can estimate the size of the output volume (output weight) during MaxPooling using the formula below.

$$W_{out} = W - F + 2P/S + 1$$

The output volume will be of the following size: $(W * W * D)$. Pooling is necessary in order for the computer to recognize an object that could appear anywhere in the image.

A layer that is fully linked has connections to both the layer before it and the layer after it. Basic dot multiplication cannot be used to calculate this. Connecting the input and output representations is this layer.

But there is a problem. Convolution is linear, but images are not. Non-linearity layers are immediately placed after the convolutional layer in order to produce a non-linear activation map.

CNN is one of the main categories which falls under neural networks when it comes to recognition of images and classification of images. Objects detection, recognition of faces, etc are some of the areas where CNN is observed being frequently used. In the first step CNN uses image classifications and then it takes an input image after that it processes that image. Upon classifying the image, it puts it under a certain category. After that a computer observes the input image as an array of pixels and it relies on the image resolution. In the process of image resolution, it will see $h * w * d$ (h = Height, w = Width, d = Dimension). e.g., An image of $9 * 9 * 3$ array of a matrix of RGB (3 refers to RGB values)

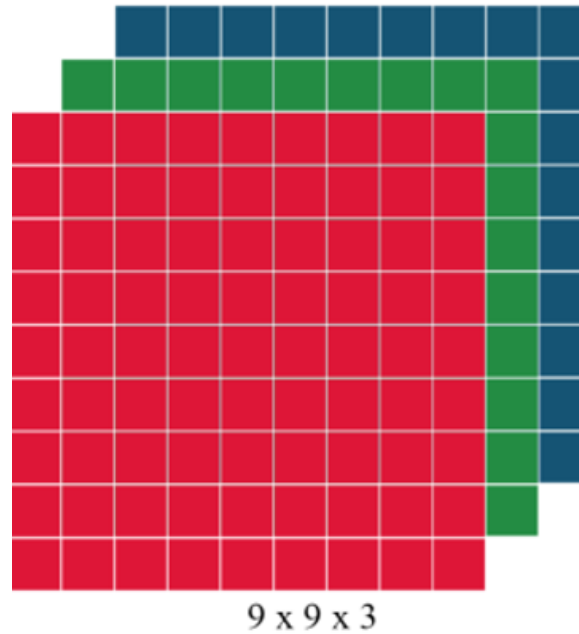


Figure 27 : Array of RGB Matrix

The test cycle initiates the development of an unavoidable arrangement to test the uncommon highlights and general usefulness on a variety of stage blends. The methodology used has its quality rigorously controlled. The method employs a pre-programmed image recognizer which has modifications to distinguish which images are skin images. Not only that the test confirms that the application is without any error but also meets all the requirements which were specified in the prerequisites report of the system

6.2 Introduction To U-Net

Deep learning has undergone a revolution thanks to the U-Net design, which was initially released in 2015. 2015's International Symposium on Biomedical Imaging cell tracking competition was decisively won by the architecture in a number of categories. They've done things like segment neural structures in transmitted light and electron microscopy stacks, among other things.

Nowadays , a standard GPU is able to perform image segmentations by using the U-net architecture where the dimensions of the image is 512x512. After going through numerous modifications and variations given its huge success. ResUnet, Attention Unet, R2Unet are a few examples which resulted from the alterations.

Even if U-Net represents a big breakthrough in deep learning, Even though U-net represents a big breakthrough when it comes to deep learning, it is rather necessary to understand how the initial approaches were used to handle such tasks. The sliding window technique, which easily won the EM segmentation competition at ISBI in 2012, was one of the key examples that came to an end. Apart from the initial

training dataset, a variety of sample patches could be produced using the sliding window method.

The U-Net architecture, however, was able to overcome two major problems with the sliding window method. The resulting patches have a lot of overlap because each pixel was taken into account separately. The result was significant overall redundancy. Another drawback was the length of the training process, which took a lot of time and resources. The following factors make it unlikely that the network will function as intended.

The U-Net is a sophisticated architecture that addresses most problems as they arise. For this strategy, fully convolutional networks are used. The U-Net's is to record both the localization and context-specific features. The design of the architecture built successfully completes this procedure. In order to produce outputs with a better resolution on the input images, the implementation's basic idea is to use successive contracting layers, which are then immediately followed by upsampling operators.[16]

Unet architecture

The architecture comprises of a symmetric expansive path that permits exact localisation and a contracting path to capture context. It performs better than the previous top technique and can be learned from very few images all the way through (a sliding window convolutional network)The typical use of convolutional network is on classification task where the idea is that we input an image and we get some class labels as output for what is actually in that image.Many visual tasks, especially in biological image processing, should include localization in the expected outcome, or the allocation of a class label to every pixel.This convolutional network brings back the sliding network approach. This network can first localize to find the class for particular pixel. Second, there are far more patches in the training data than there are training images. This is an elegant architecture, the so-called “fully convolutional network”

At the top left there is input image and in the top right there is output segmentation map. First the input goes through the contraction path where the input size is down sampled and then it has this expansive path where the image is up sampled. In between the contraction path and expansive path there is skip connections. At first the input is 572x 572 and it has a single channel. The reason behind that is that it is greyscaled and the corresponding output is 388x388 which has two channels because it has two classes. The output is not the same size as input size. Input size is much larger because padding is used there. The contracting path adheres to the standard convolutional network architecture. It consist of two 3x3 valid convolutions (unpadding convolutions) which is why here the size is reduced and it is done two times while making the channel numbers to 64. Then stride 2 downsampling is used so by using the 2x2 max pooling the input is acquired. Again two 3x3 valid convolutions are used and the number of channels are doubled until they finally reach the expansive upsampling path. So now the image is upsampled using a transpose convolution and so skip connections are used and they are concatenated along the channels dimension so half part comes from skip connection and half part

comes from upsampling. Then similar approach is used in expansive path. Here 2x2 convolution (“up-convolution”) is used and it is upsampled. It is also concatenated along with the skip connections for every single upsampled. Finally they use two 3x3 convolution again and at the final layer a 1x1 convolution layer is used. It doesn’t change the input size in any way but it changes the channel to whatever class they have. The actual size in the contraction path is square of 136 and skip connection is used the actual size in the expansive path becomes square of 104. Here the image of contraction path is cropped to match with the size of expansive path and similarly for others.

Here the fundamental idea is to add additional layers on top of a typical contracting network, replacing the pooling operators with upsampling operators. As a result, the output’s resolution is increased by these layers. High resolution characteristics from the contracting path are merged with the upsampled output for localization.[26]

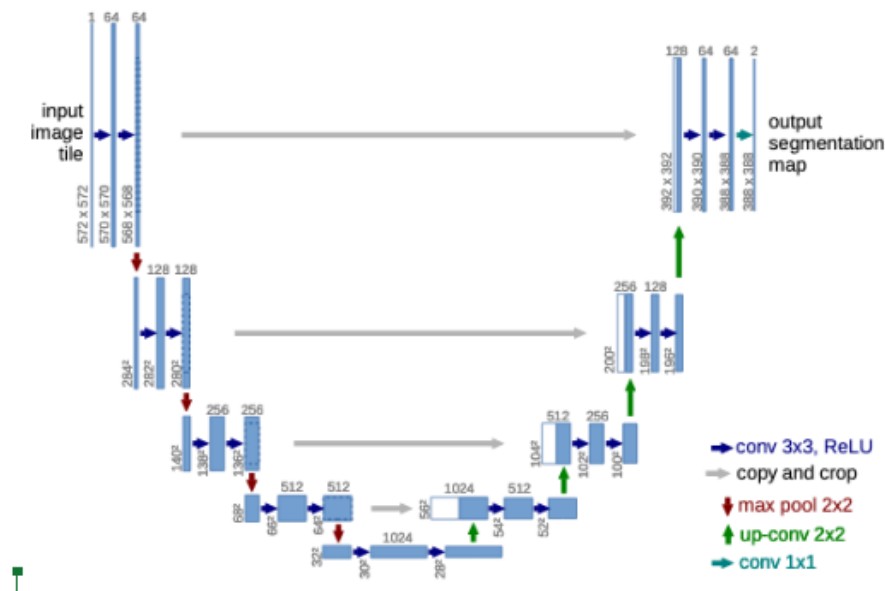


Figure 28 : U-net architecture

TensorFlow Implementation of U-Net

We will examine the TensorFlow implementation of the U-Net architecture in this section. We have build our U-Net architecture from scratch after importing all the necessary libraries.

1.Importing the required libraries: We will make use of the TensorFlow deep learning framework, as was earlier explained, to construct the U-Net architecture. Due to the fact that TensorFlow model structures now include the Keras framework, we will import both the TensorFlow library and it for this purpose. From our prior knowledge of the U-Net architecture, we know that the convolutional layer, the max-pooling layer, an input layer, and the activation function ReLU for the fundamental modeling structure are some of the essential imports. The Conv2DTranspose layer, for example, will upsample the decoder blocks we want to use, therefore we will

require some extra layers after that. Additionally, we'll combine the required skip connections using the Concatenate layers and apply the Batch Normalization layers to stabilize the training process.

2. Building the Convolution Block: We proceeded with creating the U-Net architecture after importing the necessary libraries. It can be either finish in a single class by declaring each parameter and value in turn, going through the procedure until the very end, or it can be accomplish in a series of iterative blocks. The latter approach is used by since, with the aid of a few blocks, it is easier for most users to comprehend the model architecture of U-Net. The convolution operation block, the encoder block, and the decoder block are the three iterative blocks we have used as illustrated in the architecture depiction. These three building components make it simple to construct the U-Net architecture.

We proceeded with creating the U-Net architecture after importing the necessary libraries. It can be either finish in a single class by declaring each parameter and value in turn, going through the procedure until the very end, or it can be accomplish in a series of iterative blocks. The latter approach is used by since, with the aid of a few blocks, it is easier for most users to comprehend the model architecture of U-Net. The convolution operation block, the encoder block, and the decoder block are the three iterative blocks we have used as illustrated in the architecture depiction. These three building components make it simple to construct the U-Net architecture.

3. Constructing the encoder and decoder blocks: Our following step is to create the encoder and decoder building blocks. Making these two functions is quite simple. The encoder architecture will employ sequential inputs, working its way down from the top layer. In the encoder function that we created, the convolutional block, or two convolutional layers, will be proceeded by the corresponding batch normalization and ReLU layers. As previously said, we quickly downsample these bits after running them through the convolution blocks. We follow the $\text{strides} = 2$ parameters and have used a max-pooling layer. Then we return the max-pooled output since we need the starting output to implement the skip connections.

The receiving inputs, the input for the skip connection, and the number of filters in the particular building block will all be provided as arguments for the decoder block. We have upsample the input entered into our model using the Conv2DTranspose layers. Concatenating the incoming input with the freshly upsampled layers will then yield the final value of the skip connections. Then we will use this combined function and do our convolutional block operation, going on to the next layer and providing this output result.

4. Construct the U-Net architecture: Because there are so many different components that need to be processed, the total structure is fairly large. Convolutional operation, encoder structure, and decoder structure are three distinct code blocks that we have use to divide up our respective functions into, making it simple to design the U-Net architecture in just a few lines of code. The input layer, which have the corresponding shapes from our input image, is used.

We gather all of the primary outputs and skip outputs after this step in order to transfer them to additional blocks. Up to the output, we have build each block in the decoder architecture as we go. Our intended product is produced with the necessary dimensions. In this instance, we have a single output node with sigmoid activation. To develop our final model, we have use the functional API modeling system. We then hand this model back to the user so they can use it to complete any task using the U-Net architecture.

5.Finalizing the Model: We have make sure that our image shapes can be divided into groups of at least 16 or multiples of 16. We didn't want to run into the divisibility of any odd number forms because we have employed four max-pooling layers throughout the down-sampling process. It would therefore be preferable to make sure that the architecture's sizes are equivalent to shapes like (48, 48), (80, 80), (160, 160), (256, 256), and (512, 512), as well as others of a like nature.

6.3 Introduction to Attention U-net

Attention in U-net is a method where only the relevant activations during training are highlighted. This helps to reduce the computational resources which are wasted on irrelevant activation and it helps to provide a better generalization of the network.

There are 2 types of attention:

Hard attention: This refers to highlighting regions which are relevant by cropping it which means crop the parts of the image where the object is present, for example if we have an image of a skin lesion, the parts of the picture where the lesion is present is cropped. When applying hard attention only particular region of an image, a patch of the image, is being focused at a time which makes it on differentiable and it requires reinforcement learning, for the particular dataset we used the network can pay attention as their are cropped images with lesions and in other cases not pay attention at all as there is no region of interest and there is not back propagation when it comes to hard attention.

Soft Attention: This refers to weighting different parts of the image which basically refers to assigning weight to different parts of the image where parts of the image with lesions are assigned large weights and and the other parts which does not contain relevant part of the lesion are assigned small weights which allows training to be done using back propagation. In soft attention when it comes to training the assigned weights are also getting trained which makes the model put more focus on the relevant regions of the images as the training progresses.To summarize soft attention assigns weights to the pixels of the image based on its relevance.

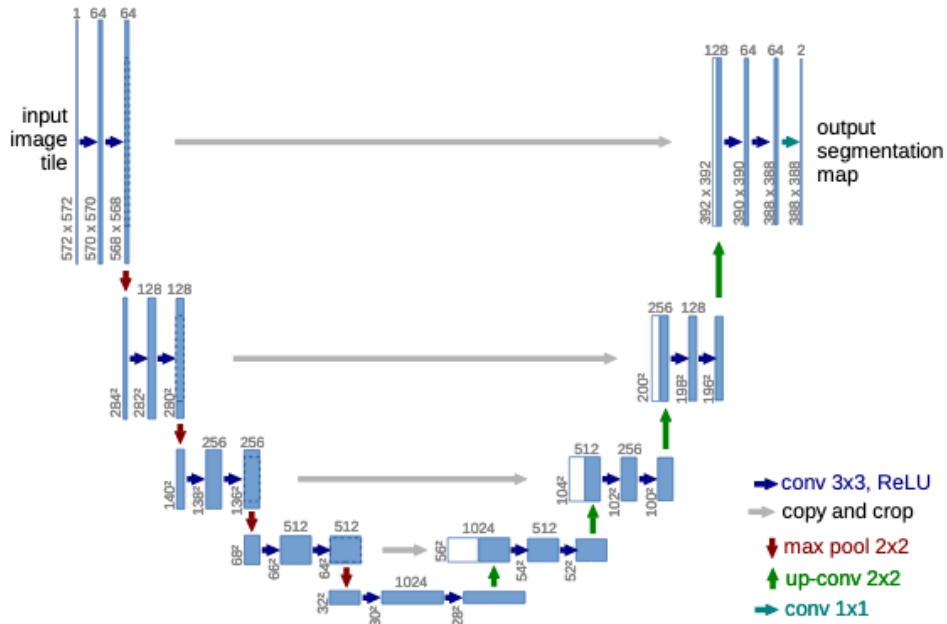


Figure 29 : Standard U-net Model

In a standard U-net model we have skip connections as on the left side there are more spatial information which provides the context when we are trying to get the layers on the right. So when we concatenate the left layer to its opposite layer to the right spatial information is combined which helps to retain good spatial information but the drawback of this is that it brings along poor feature representation from the initial layers. To overcome this soft attention is implemented at the skip connections which will help to actively suppress activations at irrelevant regions.[8]

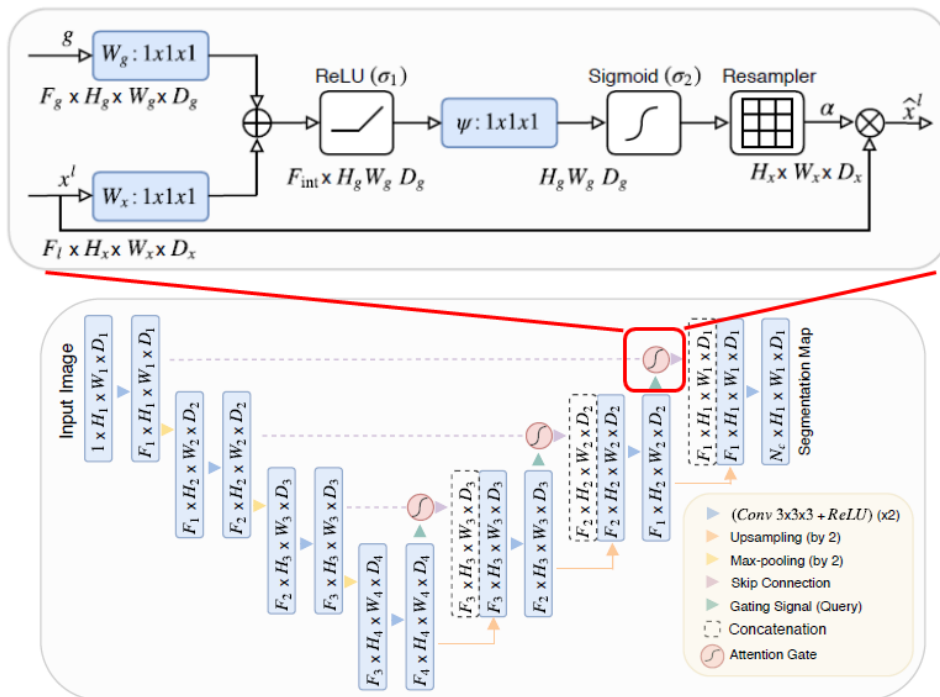


Figure 30 : Attention Gates Expanded

Attention gates have 2 inputs, one is x and the other one is called g which is the gating signal and comes from the next lowest layer of the network. Since g is coming from a deeper part of the network it has a better feature representation. On the other hand x comes from the skip connection and since it is coming from the early layers it has better spatial information.

6.4 Introduction to ResUNET

Deep Residual UNET is referred to as ResUNET. A entirely convolutional neural network called ResUNET is designed to provide excellent performance with few parameters. It improves on the UNET architecture that is already in place. Both the UNET architecture and Deep Residual Learning are used in ResUNET. In the field of remote sensing image analysis, it was first used for the roadway extraction from the elevated aerial pictures. Afterwards, it was implemented by researchers for several new purposes, including the segmentation of polyps, brain tumors, human images, and many more.[16:26][22]

6.5 Introduction to R2U-net

Recent years have seen state-of-the-art performance from deep learning (DL) based semantic segmentation algorithms. More specifically, these methods have been used to classify, segment, and identify medical images with success. U-Net, one deep learning method, has grown to be one of the most well-liked for these applications. Recurrent U-Net and Recurrent Residual U-Net, often known as RU-Net and R2U-Net, are the models we suggest. U-Net, Residual Networks, and Recurrent Convolutional Neural Networks are used to great effect in the presented models (RCNNs). These suggested structures for segmentation tasks provide a number of benefits. First, while training deep architectures, a residual unit is helpful. Second, superior feature representation for segmentation tasks is ensured by feature accumulation using recurrent residual convolutional layers. Thirdly, it enables us to create U-Net architectures that perform better for medical picture segmentation while using the same number of network parameters. Three benchmark datasets, including those for skin cancer, lung lesions, and the segmentation of blood vessels in retinal pictures, are used to test the suggested models. According to the experimental data, SegNet, U-Net, and the residual U-Net, three different fully connected convolutional neural networks (FCNs), perform better on segmentation tasks than equivalent models (ResU-Net). [7]

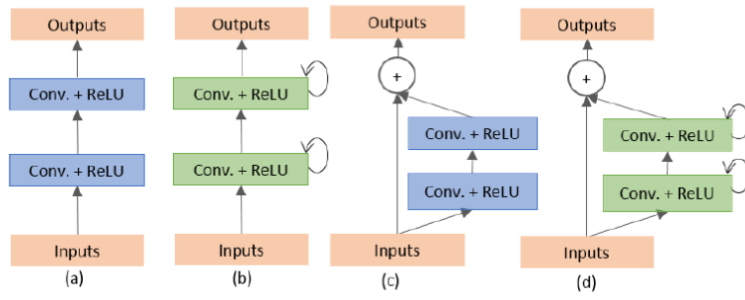


Figure 31 : Basic convolutional units of a): U-Net, (b):RU-Net, (c):Residual U-Net, (d): R2U-Net.

Chapter 7

Model Evaluation and Result Analysis

7.1 Fitting Model

For the model fitting we used a `model.fit()` function which fits the model into the `x_train` which carries the images and `y_train` which carries all the labels. The function is called and the parameters passed are the `x_train`, `y_train`, the `batch_size` which was kept at 8 for all training purposes and `validation_data=(x_validate, y_validate)`. Steps per epoch for training was derived from the total number of images in the dataset divided by batch size and the epoch was kept at 50 or 60 depending on the dataset as we used an unbalanced dataset for all the model training.

7.2 Transfer Learning Using One Pre-trained Model

Transfer Learning with CNN



Figure 7.1 :Accuracy of CNN



Figure 7.2 :Loss of CNN

```
[0. 0. 1. 0. 0. 0. 0.]
('bkl', 'benign keratosis-like lesions')
[0. 0. 1. 0. 0. 0. 0.]
('bkl', 'benign keratosis-like lesions')
[0. 0. 0. 0. 0. 0. 0.]
('akiec', 'Actinic keratoses and intraepithelial carcinomae')
[0. 0. 1. 0. 0. 0. 0.]
('bkl', 'benign keratosis-like lesions')
[0. 0. 0. 0. 0. 0. 1.]
('mel', 'melanoma')
[0. 0. 0. 0. 0. 1. 0.]
('vasc', 'vascular lesions')
[0. 0. 0. 1. 0. 0. 0.]
('df', 'dermatofibroma')
[1. 0. 0. 0. 0. 0. 0.]
('akiec', 'Actinic keratoses and intraepithelial carcinomae')
[0. 0. 0. 1. 0. 0. 0.]
('df', 'dermatofibroma')
367/367 [=====] - 8s 23ms/step - loss: 0.1459 - accuracy: 0.7199
Accuracy = 71.98891830444336 %
```

Figure 7.3 :Result of CNN

Transfer Learning with Unet

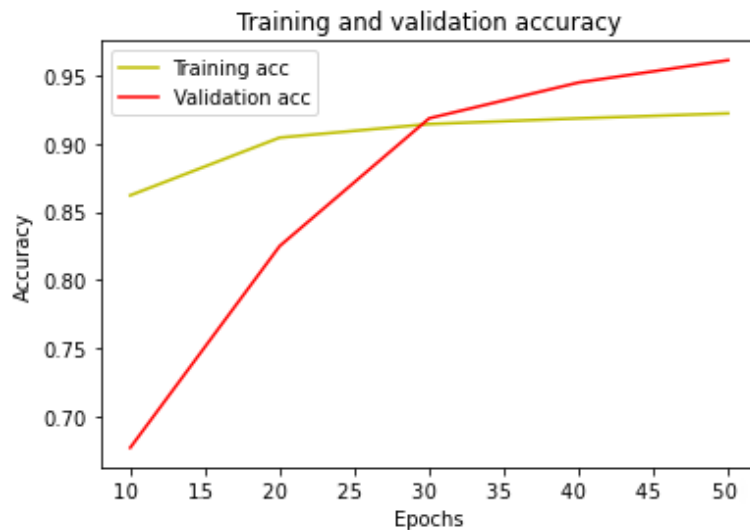


Figure 7.4 :Accuracy of Unet

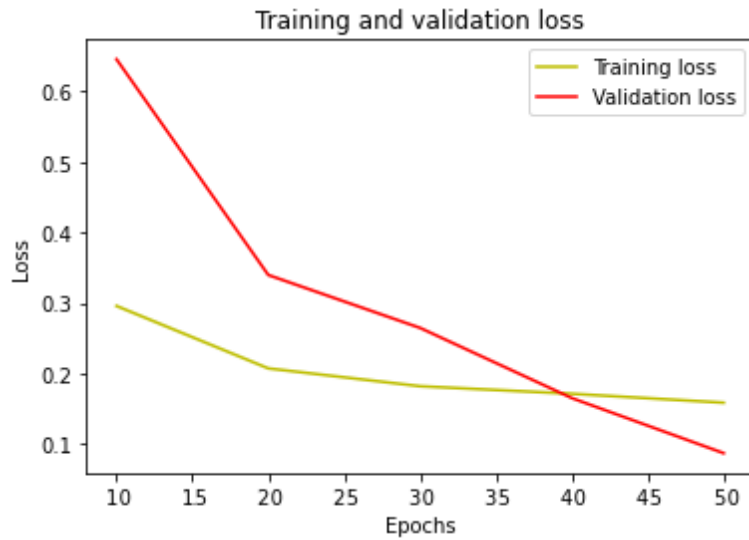


Figure 7.5 :Loss of Unet

```
In [14]: y_pred=model.predict(X_test)
...: y_pred_thresholded = y_pred > 0.5
...:
...: intersection = np.logical_and(y_test, y_pred_thresholded)
...: union = np.logical_or(y_test, y_pred_thresholded)
...: iou_score = np.sum(intersection) / np.sum(union)
...: print("IoU score is: ", iou_score)
1/1 [=====] - 1s 1s/step
IoU score is: 0.7264207658828318
```

Figure 7.6 :Result of Unet

Transfer Learning of Attention Unet



Figure 7.7 :Accuracy of Attention Unet

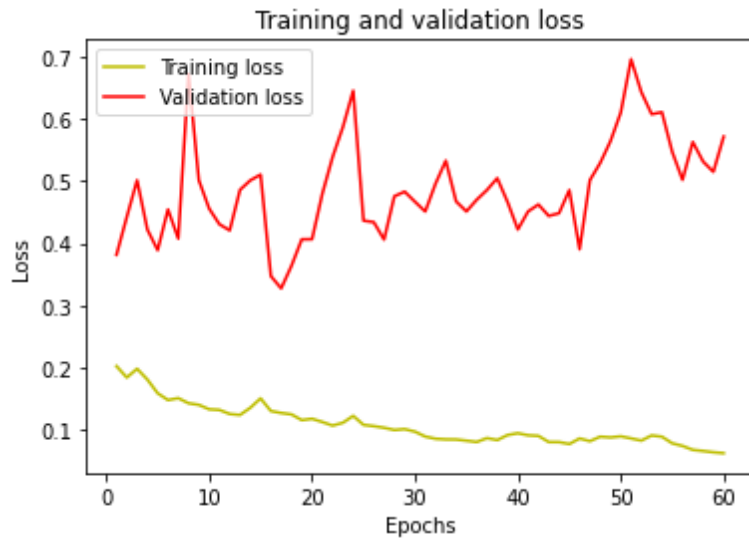


Figure 7.8 :Loss of Attention Unet

```
In [21]:
...: y_pred=model3.predict(X_test)
...: y_pred_thresholded = y_pred > 0.5
...:
...: intersection = np.logical_and(y_test, y_pred_thresholded)
...: union = np.logical_or(y_test, y_pred_thresholded)
...: iou_score = np.sum(intersection) / np.sum(union)
...: print("IoU score is: ", iou_score)
1/1 [=====] - 3s 3s/step
IoU score is: 0.7713337877755056
```

Figure 7.9 :Result of Attention Unet

Transfer Learning of ResUnet



Figure 7.10 :Accuracy of ResUnet

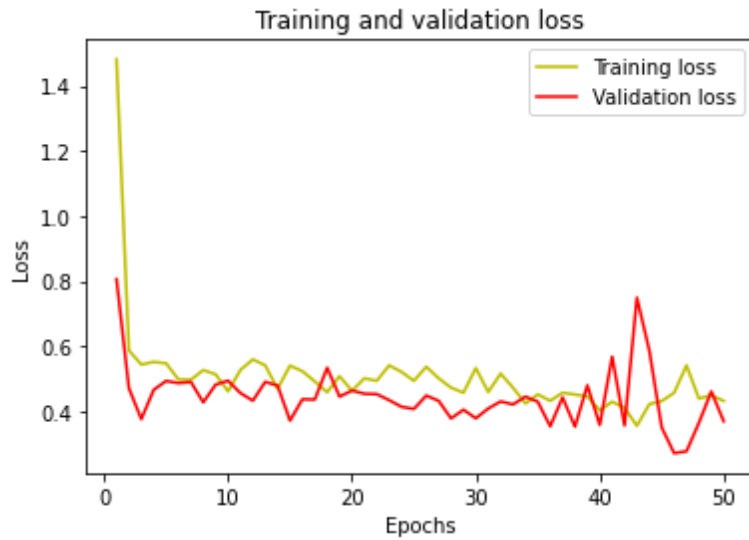


Figure 7.11 :Loss of ResUnet

```
In [11]: y_pred=model0.predict(X_test)
...: y_pred_thresholded = y_pred > 0.5
...:
...: intersection = np.logical_and(y_test, y_pred_thresholded)
...: union = np.logical_or(y_test, y_pred_thresholded)
...: iou_score = np.sum(intersection) / np.sum(union)
...: print("IoU socre is: ", iou_score)
1/1 [=====] - 1s 1s/step
IoU socre is: 0.8721282958984375
```

Figure 7.12 :Result of ResUnet

Transfer Learning of R2Unet

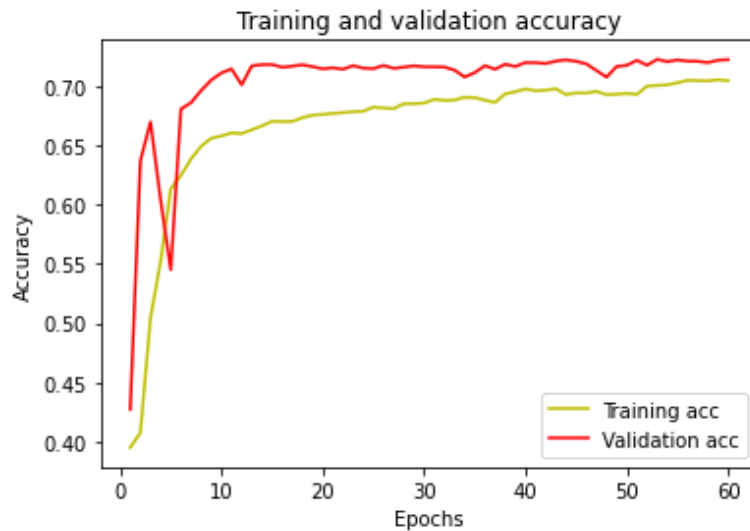


Figure 7.13 :Accuracy of R2Unet

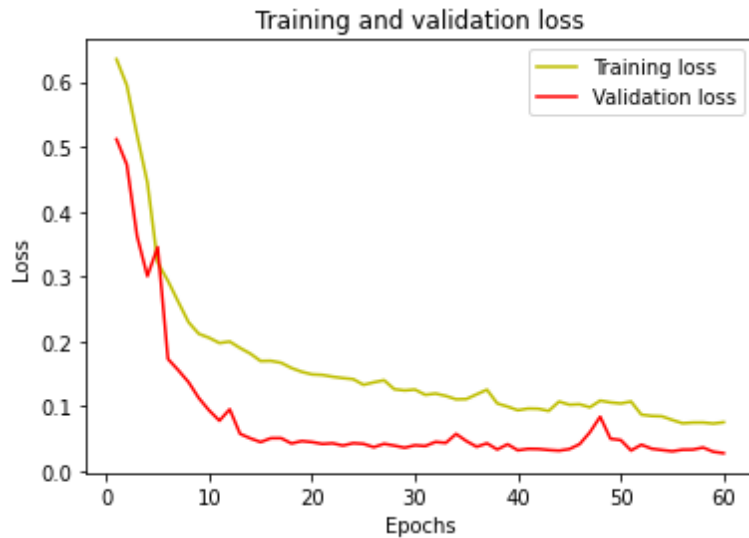


Figure 7.14 :Loss of R2Unet

```
In [10]: y_pred=model.predict(X_test)
...: y_pred_thresholded = y_pred > 0.5
...:
...: intersection = np.logical_and(y_test, y_pred_thresholded)
...: union = np.logical_or(y_test, y_pred_thresholded)
...: iou_score = np.sum(intersection) / np.sum(union)
...: print("IoU score is: ", iou_score)
1/1 [=====] - 0s 84ms/step
IoU score is: 0.748697488852382

In [11]:
```

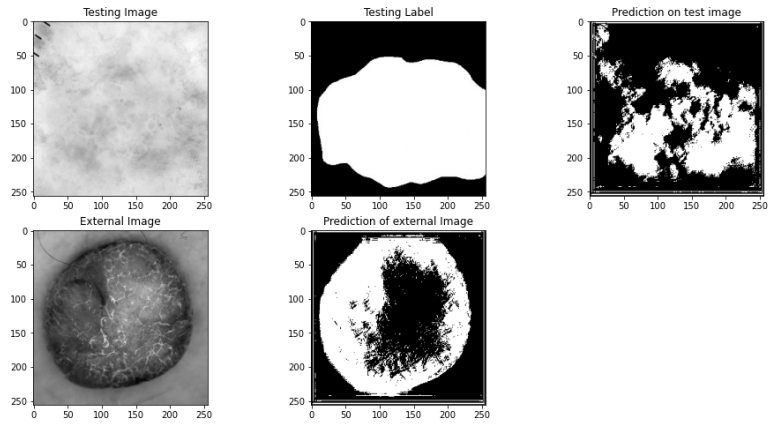
Figure 7.15 :Result of R2Unet

7.3 Analysis and Comparison of the Models

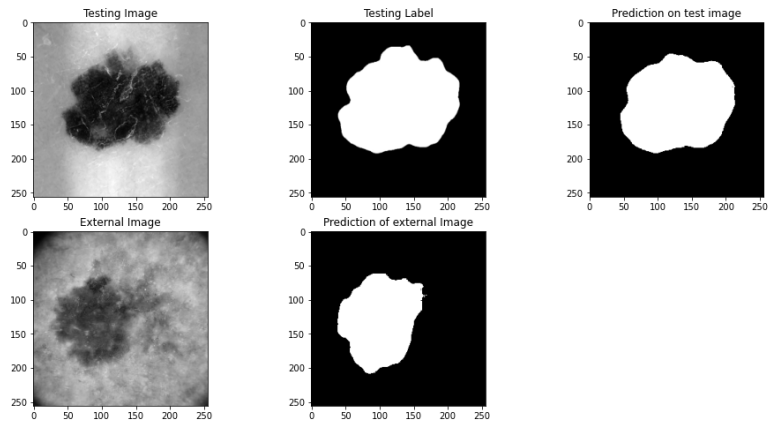
| Models | IOU/Accuracy |
|----------------|--------------|
| CNN | 71.99% |
| Unet | 72.65% |
| Attention Unet | 77.13% |
| ResUnet | 87.21% |
| R2Unet | 74.86% |

Table 7.1: Test accuracy and loss Comparison of all models

The table above shows the IOU scores for all the unet models , IoU means intersection over union, a metric used to evaluate Deep Learning algorithms by estimating how well a predicted mask or bounding box matches the ground truth data and for the CNN model test accuracy is taken into consideration for comparison. Batch size and image size were kept the same as we used the HAM10000 dataset. The highest IOU score was achieved by the ResUnet a , IOU score = 87.21 (rounded to 4S.F) and lowest score was obtained from the basic CNN models.



Test result of Unet



Test result of ResUnet

Here it is seen that the predicted images of 2 Unet models which provided predicted images with highest and lowest accuracy, ResUnet provided predicted image with the highest accuracy and standard Unet provided predicted image with the lowest accuracy.

Chapter 8

Conclusion

The aim of our study was to gather extensive data with various techniques for training amongst CNN, U-net and U-net based models like ResUnet, R2U-net, Attention U-net. We have discovered that among these models ResUnet gives the highest accuracy which is 87.21% and CNN gives the lowest accuracy which is 71.99%. The database we used had 10,015 photos, of which we deleted duplicates and split the remaining images in an 80:20 ratio to create a test-train split. Nonetheless, this study may pave the way for future model comparison and analysis studies to raise standards and produce better outcomes not only in the image processing field but also in other areas that call for CNN, U-net and machine learning. With the improvement of categorization between these models, we expect to develop a model whose outcomes are trustworthy enough to be used by physicians and healthcare facilities to support practical applications and lessen troubles for human.

Bibliography

- [1] C.-Y. Chang, H.-Y. Liao, *et al.*, “Automatic facial spots and acnes detection system,” *Journal of Cosmetics, Dermatological Sciences and Applications*, vol. 3, no. 01, p. 28, 2013.
- [2] A. H. Abdulnabi, G. Wang, J. Lu, and K. Jia, “Multi-task cnn model for attribute prediction,” *IEEE Transactions on Multimedia*, vol. 17, no. 11, pp. 1949–1959, 2015.
- [3] M. H. Jafari, N. Karimi, E. Nasr-Esfahani, *et al.*, “Skin lesion segmentation in clinical images using deep learning,” in *2016 23rd International conference on pattern recognition (ICPR)*, IEEE, 2016, pp. 337–342.
- [4] S. Kalouche, A. Ng, and J. Duchi, “Vision-based classification of skin cancer using deep learning,” *2015, conducted on Stanfords Machine Learning course (CS 229) taught*, 2016.
- [5] S. K. Patnaik, M. S. Sidhu, Y. Gehlot, B. Sharma, and P. Muthu, “Automated skin disease identification using deep learning algorithm,” *Biomedical & Pharmacology Journal*, vol. 11, no. 3, p. 1429, 2018.
- [6] P. Tschandl, C. Rosendahl, and H. Kittler, “The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions,” *Scientific data*, vol. 5, no. 1, pp. 1–9, 2018.
- [7] M. Zahangir Alom, M. Hasan, C. Yakopcic, T. M. Taha, and V. K. Asari, “Recurrent residual convolutional neural network based on u-net (r2u-net) for medical image segmentation,” *arXiv e-prints*, arXiv–1802, 2018.
- [8] C. Kaul, S. Manandhar, and N. Pears, “Focusnet: An attention-based fully convolutional network for medical image segmentation,” in *2019 IEEE 16th international symposium on biomedical imaging (ISBI 2019)*, IEEE, 2019, pp. 455–458.
- [9] A. A. Mohamed, W. A. Mohamed, and A. H. Zekry, “Deep learning can improve early skin cancer detection,” *International Journal of Electronics and Telecommunications*, vol. 65, no. 3, pp. 507–512, 2019.
- [10] A. A. Nugroho, I. Slamet, and Sugiyanto, “Skins cancer identification system of ham10000 skin cancer dataset using convolutional neural network,” in *AIP Conference Proceedings*, AIP Publishing LLC, vol. 2202, 2019, p. 020 039.
- [11] M. Vijayalakshmi, “Melanoma skin cancer detection using image processing and machine learning,” *International Journal of Trend in Scientific Research and Development (IJTSRD)*, vol. 3, no. 4, pp. 780–784, 2019.

- [12] J. K. Winkler, C. Fink, F. Toberer, *et al.*, “Association between surgical skin markings in dermoscopic images and diagnostic performance of a deep learning convolutional neural network for melanoma recognition,” *JAMA dermatology*, vol. 155, no. 10, pp. 1135–1141, 2019.
- [13] M. Z. Alom, T. Aspiras, T. M. Taha, and V. K. Asari, “Skin cancer segmentation and classification with improved deep convolutional neural network,” in *Medical Imaging 2020: Imaging informatics for healthcare, research, and applications*, SPIE, vol. 11318, 2020, pp. 291–301.
- [14] N. Gouda and J. Amudha, “Skin cancer classification using resnet,” in *2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA)*, IEEE, 2020, pp. 536–541.
- [15] R. Hollandi, Á. Diósi, G. Hollandi, N. Moshkov, and P. Horváth, “Annotatorj: An imagej plugin to ease hand annotation of cellular compartments,” *Molecular biology of the cell*, vol. 31, no. 20, pp. 2179–2186, 2020.
- [16] Z. A. Nazi and T. A. Abir, “Automatic skin lesion segmentation and melanoma detection: Transfer learning approach with u-net and dcnn-svm,” in *Proceedings of international joint conference on computational intelligence*, Springer, 2020, pp. 371–381.
- [17] S. Poorna, M. Reddy, N. Akhil, *et al.*, “Computer vision aided study for melanoma detection: A deep learning versus conventional supervised learning approach,” in *Advanced Computing and Intelligent Engineering*, Springer, 2020, pp. 75–83.
- [18] V. Srividhya, K. Sujatha, R. Ponmagal, G. Durgadevi, L. Madheshwaran, *et al.*, “Vision based detection and categorization of skin lesions using deep learning neural networks,” *Procedia Computer Science*, vol. 171, pp. 1726–1735, 2020.
- [19] V. Srividhya, K. Sujatha, R. Ponmagal, G. Durgadevi, L. Madheshwaran, *et al.*, “Vision based detection and categorization of skin lesions using deep learning neural networks,” *Procedia Computer Science*, vol. 171, pp. 1726–1735, 2020.
- [20] A. Adegun and S. Viriri, “Deep learning techniques for skin lesion analysis and melanoma cancer detection: A survey of state-of-the-art,” *Artificial Intelligence Review*, vol. 54, no. 2, pp. 811–841, 2021.
- [21] M. S. Ali, M. S. Miah, J. Haque, M. M. Rahman, and M. K. Islam, “An enhanced technique of skin cancer classification using deep convolutional neural network with transfer learning models,” *Machine Learning with Applications*, vol. 5, p. 100 036, 2021.
- [22] P. N. R. Bodavarapu, P. Srinivas, P. Mishra, V. N. Mandhala, and H.-j. Kim, “Optimized deep neural model for cancer detection and classification over resnet,” in *Smart Technologies in Data Science and Communication*, Springer, 2021, pp. 267–280.
- [23] K. Das, C. J. Cockerell, A. Patil, *et al.*, “Machine learning and its application in skin cancer,” *International Journal of Environmental Research and Public Health*, vol. 18, no. 24, p. 13 409, 2021.

- [24] M. Dildar, S. Akram, M. Irfan, *et al.*, “Skin cancer detection: A review using deep learning techniques,” *International journal of environmental research and public health*, vol. 18, no. 10, p. 5479, 2021.
- [25] R. Garg, S. Maheshwari, and A. Shukla, “Decision support system for detection and classification of skin cancer using cnn,” in *Innovations in Computational Intelligence and Computer Vision*, Springer, 2021, pp. 578–586.
- [26] D. Harrison, F. C. De Leo, W. J. Gallin, F. Mir, S. Marini, and S. P. Leys, “Machine learning applications of convolutional neural networks and unet architecture to predict and classify demosponge behavior,” *Water*, vol. 13, no. 18, p. 2512, 2021.
- [27] M. D. Alahmadi, “Multiscale attention u-net for skin lesion segmentation,” *IEEE Access*, vol. 10, pp. 59 145–59 154, 2022.
- [28] M. Nawaz, T. Nazir, M. Masood, *et al.*, “Melanoma segmentation: A framework of improved densenet77 and unet convolutional neural network,” *International Journal of Imaging Systems and Technology*, 2022.
- [29] S. Parshionikar, R. Koshy, A. Sheikh, and G. Phansalkar, “Skin cancer detection and severity prediction using computer vision and deep learning,” in *Second International Conference on Sustainable Technologies for Computational Intelligence*, Springer, 2022, pp. 295–304.
- [30] N. S. Punn and S. Agarwal, “Modality specific u-net variants for biomedical image segmentation: A survey,” *Artificial Intelligence Review*, pp. 1–45, 2022.