

# Identification of Skin Disease Using Deep Learning Model

Savita Havalkod<sup>1</sup>, Seema C K<sup>2</sup>

<sup>1</sup>MTEch Student, Maharaja Institute of Technology, Mysore

<sup>2</sup>Professor, Maharaja Institute of Technology, Mysore

## Abstract

Busy lifestyles these days have led people to forget to drink water regularly which results in inadequate hydration and oily skin, oily skin has become one of the main factors for Acne vulgaris. Acne vulgaris, particularly on the face, greatly affects a person's social, mental wellbeing and personal satisfaction for teens. Besides the fact that acne is well known as an inflammatory disorder, it was reported to have caused serious long-term consequences such as depression, scarring, mental illness, including pain and suicide. In the present work a deep learning model is designed for the identification of skin diseases.

**Keywords:** DL, Skin disease

## INTRODUCTION

Skin being the biggest organ in the human body it is imperative to keep up a sound skin. It is said that most regular skin diseases in 2013 were acne vulgaris, dermatitis, urticarial and psoriasis and it is a fact that skin illnesses have become regular around the world [1]. Acne vulgaris, particularly on the face, greatly affects a patient's passionate, social, and mental wellbeing, and personal satisfaction for youths and youthful grown-ups. Even though acne is well known as an inflammatory disease for a long time, severe outcomes of the disease such as distress, scarring, mental unsettling influences including despondency and suicide have been identified quite recently [1]. It can be demonstrated that treating acne can give a huge impact on diminishing the psychological and social pain of patients [2]. It is very important to identify the exact severity stage of acne to treat acne properly [3]. 9.4% of the global population have acne [4], but if we take it another way roughly 85% of the population in the age range from 11 years to 30 years of old have had acne at some point in their life [5].

If we take an aggregate of all the acne treatment costs in the world, each year it exceeds one billion US dollars [6] [7]. Acne wounds can be categorized into several types [8]. They are mainly blackheads, whiteheads, papules, pustules, nodules and cysts [9].



**Figure 1.1(a): Acne and Rosacea**



**Figure 1.1(b): Actinic Keratosis**



**Figure 1.1(c): Atopic Dermatitis**



**Figure 1.1(d): Bullous Disease**

**Figure 1.1: Different types of skin diseases**

## 1.1 Different Types of Skin Diseases

### 1.1.1 Acne and Rosacea

A condition that causes redness and often small, red, pus-filled bumps on the face. Rosacea most commonly affects middle-aged women with fair skin. It can be mistaken for acne or other skin conditions. Key symptoms are facial redness with swollen red bumps and small visible blood vessels. Treatments such as antibiotics or anti-acne medication can control and reduce symptoms. Left untreated, it tends to worsen over time.

Anyone can get rosacea, but it is more common among these groups:

- Middle-aged and older adults.
- Women, but when men get it, it tends to be more severe.

- People with fair skin, but it may be underdiagnosed in darker skinned people because dark skin can mask facial redness.
- People with a family history of rosacea may be at increased risk of the condition, but more research is needed to understand the role played by genetics.

Most people only experience some of the symptoms of rosacea, and the pattern of symptoms varies among individuals. While the condition is chronic (long lasting), rosacea often cycles between flare-ups and periods of remission (lack of symptoms).

#### **The symptoms of rosacea include:**

- Facial redness: This may start as a tendency to flush or blush, but over time redness may persist for longer periods. It is sometimes accompanied by a sense of tingling or burning, and the reddened skin may turn rough and scaly.
- Rash: Areas of facial redness can develop red or pus-filled bumps and pimples that resemble acne
- Visible blood vessels: These typically appear as thin red lines on the cheeks and nose.
- Skin thickening: The skin may thicken, especially on the nose, giving the nose an enlarged and bulbous appearance. This is one of the more severe symptoms, and it mostly affects men.
- Eye irritation: In what is termed ocular rosacea, the eyes become sore, red, itchy, watery, or dry. They may feel gritty or as if there is something in them, such as an eyelash. The eyelids may swell and become red at the base of the eyelashes may develop. It is important to see a health care provider if you have eye symptoms because if left untreated, eye damage and loss of vision can result.

Sometimes rosacea follows a progression, going from temporary flushing of the nose and cheeks, to longer lasting flushing, then to the appearance of a rash and small blood vessels beneath the skin. If left untreated, the skin may thicken and enlarge, leading to firm, red bumps, especially on the nose. The condition usually affects the centre of the face, but in rare cases it can extend to other parts of the body, such as the sides of the face, the ears, neck, scalp, and chest.

Scientists do not know what causes rosacea, but there are a number of theories. They know that inflammation contributes to some of the key symptoms, such as skin redness and rash, but they do not fully understand why inflammation occurs. It may in part be due to the heightened skin sensitivity in people with rosacea, to environmental stressors, such as ultraviolet (UV) light, and TO microbes that inhabit the skin. Both genetic and environmental (non genetic) factors likely play a role in the development of rosacea.

#### **1.1.2 Actinic Keratosis**

An actinic keratosis (ak-TIN-ik ker-uh-TOE-sis) is a rough, scaly patch on the skin that develops from years of sun exposure. It's often found on the face, lips, ears, forearms, scalp, neck or back of the hands.

Actinic keratosis are scaly spots or patches on the top layer of skin. With time they may become hard with a wart like surface. Also known as a solar keratosis, an actinic keratosis grows slowly and usually first appears in people over 40. You can reduce your risk of this skin condition by minimizing

your sun exposure and protecting your skin from ultraviolet (UV) rays. Left untreated, the risk of actinic keratosis turning into a type of skin cancer called squamous cell carcinoma is about 5% to 10%.

Actinic keratosis vary in appearance. Symptoms include:

- Rough, dry or scaly patch of skin, usually less than 1 inch (2.5 centimetres) in diameter
- Flat to slightly raised patch or bump on the top layer of skin
- In some cases, a hard, wart like surface Colour variations, including pink, red or brown Itching, burning, bleeding or crusting
- New patches or bumps on sun-exposed areas of the head, neck, hands and forearms Apricot

It can be difficult to distinguish between noncancerous spots and cancerous ones. So it's best to have new skin changes evaluated by a health care provider — especially if a scaly spot or patch persists, grows or bleeds.

### 1.1.3 Atopic Dermatitis

Atopic dermatitis, often referred to as eczema, is a chronic (long-lasting) disease that causes inflammation, redness, and irritation of the skin. It is a common condition that usually begins in childhood; however, anyone can get the disease at any age. Atopic dermatitis is not contagious, so it cannot be spread from person to person. Atopic dermatitis causes the skin to become extremely itchy. Scratching leads to further redness, swelling, cracking, “weeping” clear fluid, crusting, and scaling. In most cases, there are periods of time when the disease is worse, called flares, followed by periods when the skin improves or clears up entirely, called remissions.

Researchers do not know what causes atopic dermatitis, but they do know that genes, the immune system, and the environment play a role in the disease. Depending on the severity and location of the symptoms, living with atopic dermatitis can be hard. Treatment can help control symptoms. For many people, atopic dermatitis improves by adulthood, but for some, it can be a lifelong illness. Atopic dermatitis is a common disease and usually appears during infancy and childhood. For many children, atopic dermatitis goes away before the teenage years. However, some children who develop atopic dermatitis may continue to have symptoms as teens and adults. Occasionally, for some people, the disease first appears during adulthood.

The chance of developing atopic dermatitis is higher if there is a family history of atopic dermatitis, hay fever, or asthma. In addition, research shows that atopic dermatitis is more common in non-Hispanic black children and that women and girls tend to develop the disease slightly more often than men and boys do. The most common symptom of atopic dermatitis is itching, which can be severe. Other common symptoms include:

- Red, dry patches of skin.
- Rashes that may ooze, weep clear fluid, or bleed when scratched.
- Thickening and hardening of the skin.
- The symptoms can flare in multiple areas of the body at the same time and can appear in the same locations and in new locations. The appearance and location of the rash vary depending on age;



however, the rash can appear anywhere on the body. Patients with darker skin tones often experience darkening or lightening of the skin in areas of skin inflammation.

During infancy and up to 2 years of age, it is most common for a red rash, which may ooze when scratched, to appear on the:

- Face
- Scalp

Area of skin around joints that touch when the joint bends. Some parents worry that the infant has atopic dermatitis in the diaper area; however, the condition rarely appears in this area. During childhood, usually 2 years of age to puberty, it is most common for a red thickened rash, which may ooze or bleed when scratched, to appear on the:

- Elbows and knees, usually in the bend.
- Neck.
- Ankles.

No one knows what causes atopic dermatitis; however, researchers know that changes in the protective layer of the skin can cause it to lose moisture. This can cause the skin to become dry, leading to damage and inflammation in the skin. New research suggests that inflammation directly triggers sensations of itch which in turn cause the patient to scratch. This leads to further damage of the skin as well as increased risk for infection with bacteria.

Researchers do know that the following may contribute to the changes in the skin barrier, which helps control moisture:

- Changes (mutations) in genes.
- Problems with the immune system.
- Exposure to certain things in the environment.

The chance of developing atopic dermatitis is higher if there is a family history of the disease, which suggests that genetics may play a role in the cause. Recently, researchers found changes to genes that control a specific protein and help our bodies maintain a healthy layer of skin. Without the normal levels of this protein, the skin barrier changes, allowing moisture to escape and exposing the skin immune system to the environment, leading to atopic dermatitis. Researchers continue to study genes to better understand how different mutations cause atopic dermatitis.

Environmental factors may trigger the immune system to change the protective barrier of the skin allowing more moisture to escape, which can lead to the atopic dermatitis. These factors may include:

- Exposure to tobacco smoke.
- Certain types of air pollutants.
- Fragrances and other compounds found in skin products and soaps.
- Excessively dry skin.

#### **1.1.4 Bullous Disease**

Bullous pemphigoid (BUL-us PEM-fih-goid) is a rare skin condition that causes large, fluid-filled blisters. They develop on areas of skin that often flex — such as the lower abdomen, upper thighs

or armpits. Bullous pemphigoid is most common in older adults. Bullous pemphigoid occurs when your immune system attacks a thin layer of tissue below your outer layer of skin. The reason for this abnormal immune response is unknown, although it sometimes can be triggered by taking certain medications.

Bullous pemphigoid often goes away on its own in a few months, but may take as many as five years to resolve. Treatment usually helps heal the blisters and ease any itching. It may include corticosteroid medications, such as prednisone, and other drugs that suppress the immune system. Bullous pemphigoid can be life-threatening, especially for older people who are already in poor health. The signs and symptoms of bullous pemphigoid may include:

- Itching skin, weeks or months before blisters form
- Large blisters that don't easily rupture when touched, often along creases or folds in the skin
- Skin around the blisters that is normal, reddish or darker than normal
- Eczema or a hive-like rash
- Small blisters or sores in the mouth or other mucous membranes (benign mucous membrane pemphigoid)

The blisters occur because of a malfunction in your immune system. Your body's immune system normally produces antibodies to fight bacteria, viruses or other potentially harmful foreign substances. For reasons that are not clear, the body may develop an antibody to a particular tissue in your body. In bullous pemphigoid, the immune system produces antibodies to the fibers that connect the outer layer of skin (epidermis) and the next layer of skin (dermis). These antibodies trigger inflammation that produces the blisters and itching of bullous pemphigoid.

Bullous pemphigoid usually appears randomly with no clear factors contributing to the onset of disease. Some cases may be triggered by:

- Medications: Prescription drugs that may cause bullous pemphigoid include etanercept (Enbrel), sulfasalazine (Azulfidine), furosemide (Lasix) and penicillin.
- Light and radiation: Ultraviolet light therapy to treat certain skin conditions may trigger bullous pemphigoid, as can radiation therapy to treat cancer.
- Medical conditions: Disorders that may trigger bullous pemphigoid include psoriasis, lichen planus, diabetes, rheumatoid arthritis, ulcerative colitis and multiple sclerosis.

## 1.2 Motivation

Skin diseases affect millions of people worldwide, and accurate and timely diagnosis is crucial for effective treatment and management. Traditional diagnosis methods often require expert dermatologists, leading to long waiting times and increased healthcare costs. Leveraging advanced technologies like Convolutional Neural Networks (CNNs) can aid in automating the diagnosis process and providing accessible and efficient healthcare solutions. By developing a CNN-based classification model for skin diseases, we can assist in early detection, improve patient outcomes, and alleviate the burden on healthcare systems.

### 1.3 Problem Statement

The problem at hand is to develop a CNN-based classification model for accurately identifying and classifying different types of skin diseases based on dermatological images. The model will take input as images of skin lesions or affected areas and provide predictions indicating the specific skin disease or condition. The goal is to create a reliable and efficient system that can assist healthcare professionals in diagnosing skin diseases, facilitating early intervention and appropriate treatment.

### 1.4 Objectives

The objectives of this project include:

1. Dataset collection: Gather a diverse and comprehensive dataset of dermatological images, covering various types of skin diseases and conditions. The dataset should include images of different skin tones, age groups, and disease severities to ensure the model's effectiveness across a wide range of scenarios.
2. Data pre-processing and augmentation: Pre-process the collected dermatological images by resizing, normalizing pixel values, and removing noise or artefacts. Apply data augmentation techniques like rotation, flipping, or zooming to augment the dataset, improve generalization, and account for variations in image acquisition.
3. CNN model architecture: Design and implement CNN architecture suitable for skin disease classification. The model should have multiple convolutional layers to extract relevant features from the input images effectively. Fine-tune the architecture and optimize hyper parameters to achieve high classification accuracy.
4. Training and evaluation: Train the CNN model using the pre-processed dataset and evaluate its performance using evaluation metrics such as accuracy, precision, recall, and F1-score. Employ techniques like cross-validation to ensure reliable performance assessment and minimize over fitting.
5. Optimization and fine-tuning: Optimize the model through techniques like hyper parameter tuning, learning rate adjustments, or regularization to enhance its performance and robustness. Iteratively refine the model based on evaluation results and domain knowledge.
6. Deployment and usability: Develop a user-friendly interface or application where users, including healthcare professionals, can upload dermatological images and obtain accurate disease classification results in real-time. Ensure the systems intuitive, provides explanations for predictions, and assists in decision-making for appropriate treatments or referrals.

By achieving these objectives, the project aims to contribute to improved access to dermatological care, faster diagnosis, and more efficient treatment of skin diseases. The automated skin disease classification system can be a valuable tool for dermatologists, primary care physicians, and individuals seeking reliable and accessible healthcare solutions.

### 1.5 Hardware and Software Requirements

#### Hardware Requirements

- ✚ RAM : 4GB
- ✚ Hard disk : 100 GB
- ✚ Process : i3 core

## Software Requirements

- IDE : Anaconda
- Language : Python

Anaconda is the data science platform for data scientists, IT professionals and business leaders of tomorrow. It is a distribution of Python, R, etc. With more than 300 packages for data science, it becomes one of the best platforms for any project. Here, one can use anaconda for python programming.

### 1.5.1 Anaconda

Anaconda is an open-source distribution for python and R. It is used for data science, machine learning, deep learning, etc. With the availability of more than 300 libraries for data science, it becomes fairly optimal for any programmer to work on anaconda for data science.



**Figure 1.2: Anaconda Open Source Tool**

Anaconda helps in simplified package management and deployment. Anaconda comes with a wide variety of tools to easily collect data from various sources using various machine learning and AI algorithms. It helps in getting an easily manageable environment setup which can deploy any project with the click of a single button.

### 1.5.2 Python

Python is a widely used general-purpose, high level programming language. It was created by Guido van Rossum in 1991 and further developed by the Python Software Foundation. It was designed with an emphasis on code readability, and its syntax allows programmers to express their concepts in fewer lines of code.

Python is a programming language that lets you work quickly and integrate systems more efficiently.

Before we start Python programming, we need to have an interpreter to interpret and run our programs. There are certain online interpreters like <https://ide.geeksforgeeks.org/> that can be used to run Python programs without installing an interpreter.

**Windows:** There are many interpreters available freely to run Python scripts like IDLE (Integrated Development Environment) that comes bundled with the Python software downloaded from <http://python.org/>.

**Linux:** Python comes preinstalled with popular Linux distros such as Ubuntu and Fedora. To check which version of Python you're running, type "python" in the terminal emulator. The interpreter should start and print the version number.

**Mac OS:** Generally, Python 2.7 comes bundled with mac OS. You'll have to manually install Python 3 from <http://python.org/>.



## LANGUAGE FEATURES

- **Interpreted**
  - There are no separate compilation and execution steps like C and C++.
  - Directly *run* the program from the source code.
  - Internally, Python converts the source code into an intermediate form called byte codes which is then translated into native language of specific computer to run it.
  - No need to worry about linking and loading with libraries, etc.
- **Platform Independent**
  - Python programs can be developed and executed on multiple operating system platforms.
  - Python can be used on Linux, Windows, Macintosh, Solaris and many more.
- **Free and Open Source:** Redistributable

## 1.6 Organization of Report

This report is organized as follows:

- Chapter 2 presents discussion on literature review
- Chapter 3 explains about Methodology and Implementation
- Chapter 4 contains results and discussion
- Chapter 5 contains conclusion and future scope

### Chapter 2

#### LITERATURE SURVEY

Under this section, the authors reviewed several research papers to recognize existing research areas building a mobile application with the possibilities of acne subtype, skin sensitivity, and acne density identification together with recommending homeopathic treatments. Further, it is subdivided into areas of convolutional neural network for image classification, Natural language processing for text classification and recommendation algorithm.

In 2016, the authors have built a mobile application to detect and classify acne. The main objective of this research was to find a proper solution to identify and classify acne severity from photos taken by a cell phone. Here three different segmentation methods have been used in which two level k-means clustering outperformed the others and also when it comes to classification part two machine learning methods were used. Here FCM (fuzzy c-means) method outperformed the Support Vector Machine (SVM). The authors have mentioned that the texture method they used is insufficient and needs further improvement. Also, they have mentioned that color tones and hair in the pictures affect accuracy. Therefore, Cureto focuses on tropical skin and to eliminate inaccuracy areas with heavy facial air will be ignored [13].

In research done by Sunyani technical university in 2019, they have used a Web-based approach to diagnose Skin Diseases. Researchers concluded that CNN is enough to extract features from the images. Also, they have successfully managed to reduce the computational time (0.0001 seconds) together with an increase in accuracy. However, the study did not specifically focus on acne subtype classification and also this is mainly a web-based application [14] therefore Cureto will address the issue of portability and subtype classification.

Recent research done in 2018 focused on identifying acne using smartphone images [11]. However, they mainly classify Acne into only two subtypes namely papules and pustules. They have used a facial recognition algorithm to separately identify features in the face and to classify acne accordingly. One of the main limitations the authors have presented is that the presence of more than one face and also bad lighting will affect the classification. However, the main disadvantage seen in this study is even though the accuracy is high they have classified acne only to two types but dermatologists and also websites have specifically stated that Acne can be of mainly six types as mentioned above.

In the classification of images based on convolutional neural networks, the authors have found many researches on the diagnosis of a medical condition based on deep learning techniques. In another study, researchers jointly train a CNN-RNN model and achieve multi label classification and common thorax disease analysis. Their experimental results demonstrate the excellent capability of deep networks for medical images to reflect features [15]. Going further, in a study done by Cardiff University [15] they have developed a grading of acne visual and Counting via Label Distribution learning. Authors conclude that Deep Feature outperforms both the other method; they have clearly stated that deep features represent acne via information of a high semantic level and better performance. Accordingly, it is also said that ResNet50 [16] reaches the best performance in basic CNN models. Therefore, in the proposed research component CNN model will be used to classify acne subtype, acne density and skin sensitivity.

Under the classification using Natural Language Processing the most relevant research to Acne identification is a study to apply natural language processing to Reddit comments about dermatology topics to assess for feasibility and potential for insights and engagement [17] which was done in 2019 using Latent Dirichlet Allocation (LDA). The biggest limitation in this as confirmed by the research team itself is the use of LDA where in which it is an unsupervised model, which means that there is no ground-based truth comparable to the model performance. However, during this research, the authors have decided to use a supervised learning model. In our study, the authors decided to use Naïve Bayes. Support Vector Machine and also adding a novelty by using a deep learning classifier namely [18] convolutional neural networks and finally selecting the most appropriate classifier according to the dataset.

Several types of research have been done related to the recommendation, but none has been found up to now specifically for acne personalized homeopathic recommendation. A study which has already done a literature review on existing medicinal recommendation systems, have described the various features those recommendation systems have used [9]. This study shows that CADRE [19] has used a Collaborative Filtering algorithm to develop a recommendation system for online pharmacies to recommend drugs. Collaborative Filtering is best suited for recommending based on how other users had responded to a certain item. The features of the item are not considered here, only user preferences are considered. The content-based Algorithm is best suited for recommending based on the features of said item that are more related to a single user [19].

## 2.1 Faster RCNN

Recent advances in object detection are driven by the success of region proposal methods and region-based convolutional neural networks (RCNNs) [5]. Although region-based CNNs were computationally expensive as originally developed in [5], their cost has been drastically reduced thanks to sharing convolutions across proposals [1], [2]. The latest incarnation, Fast R-CNN [2], achieves near real-time rates using very deep networks [3], when ignoring the time spent on region proposals. Now, proposals are the test-time computational bottleneck in state-of-the-art detection systems.

Region proposal methods typically rely on inexpensive features and economical inference schemes. Selective Search, one of the most popular methods, greedily merges super pixels based on engineered low-level features. Yet when compared to efficient detection networks, Selective Search is an order of magnitude slower, at 2 seconds per image in a CPU implementation. Edge Boxes currently provides the best trade-off between proposal quality and speed, at 0.2 seconds per image. Nevertheless, the region proposal step still consumes as much running time as the detection network.

One selective search method required much running time for detections of object location [8]. Faster R-CNN overcome this challenge with introducing the Region Proposal Network (RPN), which handshake convolution features with the classification network and two networks are joined as one network that can trained and tested by end-to-end process [4]. The use of this process the running time for region proposal generation take small time and this framework can maintain the detection rate time and better perform the state-of-the-art in object detection accuracy using CNN Model [4].

The machine learning based support vector machine (SVM) algorithm used for identification of apple skin disease. The characteristics of skin like colour, shape and texture extracted from skin images. Now support vector machine (SVM) finds the automatic rapid diagnosis of apple Skin disease [22]. The algorithm is more effective and feasible to recognition of apple skin disease, with accuracy is above 90% on the apple skin disease images dataset. Image Dataset contained 2462 images.

## 2.2 Artificial Neural Network

Artificial Neural Network is used to find and predict the nutrition available in the crops. The ANN can predict the complex mapping when reliable dataset available, same fed for methods. The input or data like humidity, temperature, cloud cover and wind direction can be useful for detection; same data get divided into two sting input and output.

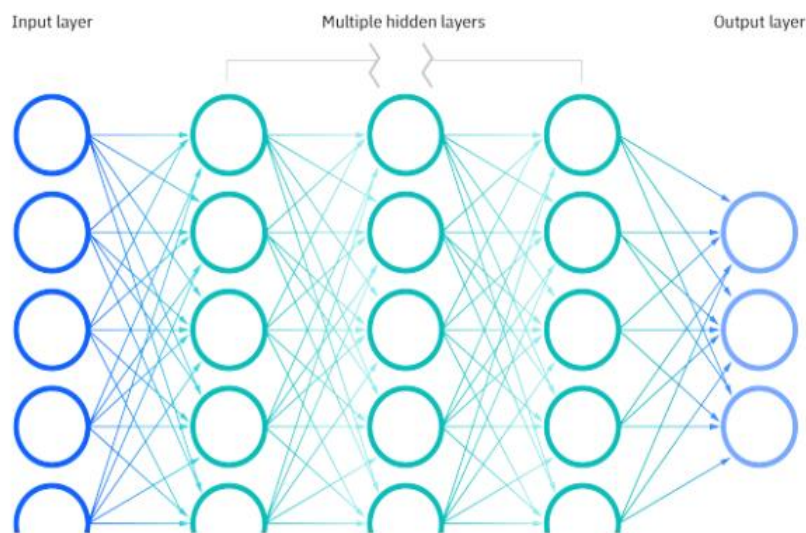
Artificial Neural Networks (ANN) are algorithms based on brain function and are used to model complicated patterns and forecast issues. The Artificial Neural Network (ANN) is a deep learning method that arose from the concept of the human brain Biological Neural Networks. The development of ANN was the result of an attempt to replicate the workings of the human brain. The workings of ANN are extremely similar to those of biological neural networks, although they are not identical. ANN algorithm accepts only numeric and structured data.

Neural networks, also known as artificial neural networks (ANNs) or simulated neural networks (SNNs), are a subset of machine learning and are at the heart of deep learning algorithms. Their name and

structure are inspired by the human brain, mimicking the way that biological neurons signal to one another.

Artificial neural networks (ANNs) as shown in Figure 2.1 are comprised of a node layers, containing an input layer, one or more hidden layers, and an output layer. Each node, or artificial neuron, connects to another and has an associated weight and threshold. If the output of any individual node is above the specified threshold value, that node is activated, sending data to the next layer of the network. Otherwise, no data is passed along to the next layer of the network.

Neural networks rely on training data to learn and improve their accuracy over time. However, once these learning algorithms are fine-tuned for accuracy, they are powerful tools in computer science and artificial intelligence, allowing us to classify and cluster data at a high velocity. Tasks in speech recognition or image recognition can take minutes versus hours when compared to the manual identification by human experts. One of the most well-known neural networks is Google’s search algorithm.



**Figure 2.1: Artificial Neural Network**

### 2.3 Neural Networks

Think of each individual node as its own linear regression model, composed of input data, weights, a bias (or threshold), and an output. The formula would look something like this:

$$\sum w_i x_i + \text{bias} = w_1 x_1 + w_2 x_2 + w_3 x_3 + \text{bias}$$

$$\text{output} = f(x) = 1 \text{ if } \sum w_1 x_1 + b \geq 0; 0 \text{ if } \sum w_1 x_1 + b < 0$$

Once an input layer is determined, weights are assigned. These weights help determine the importance of any given variable, with larger ones contributing more significantly to the output compared to other inputs. All inputs are then multiplied by their respective weights and then summed. Afterward, the output is passed through an activation function, which determines the output. If that output exceeds a given threshold, it “fires” (or activates) the node, passing data to the next layer in the network. This

results in the output of one node becoming in the input of the next node. This process of passing data from one layer to the next layer defines this neural network as a feed forward network.

Let's break down what one single node might look like using binary values. We can apply this concept to a more tangible example, like whether you should go surfing (Yes: 1, No: 0). The decision to go or not to go is our predicted outcome, or  $\hat{y}$ . Let's assume that there are three factors influencing your decision-making:

Are the waves good? (Yes: 1, No: 0)

Is the line-up empty? (Yes: 1, No: 0)

Has there been a recent shark attack? (Yes: 0, No: 1)

Then, let's assume the following, giving us the following inputs:

$X_1 = 1$ , since the waves are pumping

$X_2 = 0$ , since the crowds are out

$X_3 = 1$ , since there hasn't been a recent shark attack

Now, we need to assign some weights to determine importance. Larger weights signify that particular variables are of greater importance to the decision or outcome.

$W_1 = 5$ , since large swells don't come around often

$W_2 = 2$ , since you're used to the crowds

$W_3 = 4$ , since you have a fear of sharks

Finally, we'll also assume a threshold value of 3, which would translate to a bias value of  $-3$ . With all the various inputs, we can start to plug in values into the formula to get the desired output.

$$\hat{Y} = (1*5) + (0*2) + (1*4) - 3 = 6$$

If we use the activation function from the beginning of this section, we can determine that the output of this node would be 1, since 6 is greater than 0. In this instance, you would go surfing; but if we adjust the weights or the threshold, we can achieve different outcomes from the model. When we observe one decision, like in the above example, we can see how a neural network could make increasingly complex decisions depending on the output of previous decisions or layers.

In the example above, we used perceptron's to illustrate some of the mathematics at play here, but neural networks leverage sigmoid neurons, which are distinguished by having values between 0 and 1. Since neural networks behave similarly to decision trees, cascading data from one node to another, having  $x$  values between 0 and 1 will reduce the impact of any given change of a single variable on the output of any given node, and subsequently, the output of the neural network.

As we start to think about more practical use cases for neural networks, like image recognition or classification, we'll leverage supervised learning, or labelled datasets, to train the algorithm. As we train the model, we'll want to evaluate its accuracy using a cost (or loss) function. This is also commonly referred to as the mean squared error (MSE). In the equation below,

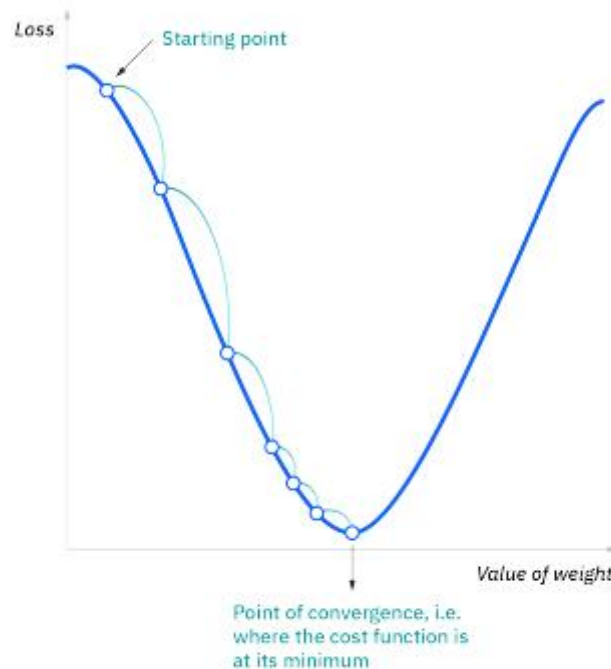


$i$  represents the index of the sample,  
 $\hat{y}$  is the predicted outcome,  
 $y$  is the actual value, and  
 $m$  is the number of samples.

$$\text{Cost Function} = \text{MSE} = \frac{1}{2m} \sum_{i=1}^m (y^{(i)} - \hat{y}^{(i)})^2$$

Ultimately, the goal is to minimize our cost function to ensure correctness of fit for any given observation. As the model adjusts its weights and bias, it uses the cost function and reinforcement learning to reach the point of convergence, or the local minimum. The process in which the algorithm adjusts its weights is through gradient descent, allowing the model to determine the direction to take to reduce errors (or minimize the cost function). With each training example, the parameters of the model adjust to gradually converge at the minimum.

Most deep neural networks are feed forward, meaning they flow in one direction only, from input to output. However, you can also train your model through back propagation; that is, move in the opposite direction from output to input. Back propagation allows us to calculate and attribute the error associated with each neuron, allowing us to adjust and fit the parameters of the model(s) appropriately as shown in Figure 2.2.



**Figure 2.2: Point of Convergence**

## 2.4 Imaging Object Technology (IOT)

Colour region and machine vision able to inspect object in light invisible such as ultraviolet (UV), near-infrared (NIR), and infrared (IR) [1]. The invisible light region provides the object information which is helpful to determining plant maturity, disease, vegetable variety, ripeness and

quality. It also useful in detecting quality and safety such as, defect, functional properties, disease and composition of plant, grain and nuts, vegetable and fruits[1]. Ultimately Imaging technology help to provide better quality and safety foods.

Image processing is the process of transforming an image into a digital form and performing certain operations to get some useful information from it. The image processing system usually treats all images as 2D signals when applying certain predetermined signal processing methods. Blind image DE convolution is the challenge of recovering a clear picture from a blurry and noisy one without knowing precisely how the image was blurred, as obtained by an ideal pinhole camera. The unidentified blurring operation might be brought on by defocus, camera movement, scene motion, or other optical defects. A trade-off between exposure duration and aperture setting is necessary for proper photography exposure. The photographer might utilize a big aperture or a lengthy exposure period when the lighting is bad. The first option produces motion blur when the camera moves concerning objects in the scene while the exposure takes place. When using the second option, things farther from the focal plane become out-of-focus blurred. This may lead to blind DE convolution in image processing.

Phase, in a nutshell, contains information about the positions of features. Phase-only and magnitude-only photos cannot be combined to produce the original. To obtain the original, multiply them in the Fourier domain and reverse the transformation. A repeated waveform's phase describes the position or timing of a particular point within a wave cycle. Instead of the actual absolute phases of the signals, the phase difference between waves usually matters. Image processing has been extensively used in medical research and has enabled more efficient and accurate treatment plans. For example, it can be used for the early detection of breast cancer using a sophisticated nodule detection algorithm in breast scans. Since medical usage calls for highly trained image processors, these applications require significant implementation and evaluation before they can be accepted for use.

Image processing can be used to recover and fill in the missing or corrupt parts of an image. This involves using image processing systems that have been trained extensively with existing photo datasets to create newer versions of old and damaged photos

## 2.5 Wireless Sensor Technology

Recent years have witnessed tremendous advances in the design and applications of wirelessly networked and embedded sensors. Wireless sensor nodes are typically low-cost, low-power, small devices equipped with limited sensing, data processing and wireless communication capabilities, as well as power supplies. They leverage the concept of wireless sensor networks (WSNs), in which a large (possibly huge) number of collaborative sensor nodes could be deployed. As an outcome of the convergence of micro-electro-mechanical systems (MEMS) technology, wireless communications, and digital electronics, WSNs represent a significant improvement over traditional sensors. In fact, the rapid evolution of WSN technology has accelerated the development and deployment of various novel types of wireless sensors, e.g., multimedia sensors. Fulfilling Moore's law, wireless sensors are becoming smaller and cheaper, and at the same time more powerful and ubiquitous.

Wireless Sensor Technologies based application mostly suitable for distributed data collecting and monitoring [23]. The WST record the parameters like climate influence, crop canopy influence, environmental changes, climate monitoring, precision agriculture, pest control and precision irrigation [23]. Which provides new features that developed agriculture economically viable.

Wireless sensor networks (WSNs) refer to networks of spatially dispersed and dedicated sensors that monitor and record the physical conditions of the environment and forward the collected data to a central location. WSNs can measure environmental conditions such as temperature, sound, pollution levels, humidity and wind.

These are similar to wireless ad hoc networks in the sense that they rely on wireless connectivity and spontaneous formation of networks so that sensor data can be transported wirelessly. WSNs monitor physical conditions, such as temperature, sound, and pressure. Modern networks are bi-directional, both collecting data and enabling control of sensor activity. The development of these networks was motivated by military applications such as battlefield surveillance. Such networks are used in industrial and consumer applications, such as industrial process monitoring and control and machine health monitoring and agriculture.

A WSN is built of "nodes" – from a few to hundreds or thousands, where each node is connected to other sensors. Each such node typically has several parts: a radio transceiver with an internal antenna or connection to an external antenna, a microcontroller, an electronic circuit for interfacing with the sensors and an energy source, usually a battery or an embedded form of energy harvesting. A sensor node might vary in size from a shoebox to (theoretically) a grain of dust, although microscopic dimensions have yet to be realized. Sensor node cost is similarly variable, ranging from a few to hundreds of dollars, depending on node sophistication. Size and cost constraints constrain resources such as energy, memory, computational speed and communications bandwidth. The topology of a WSN can vary from a simple star network to an advanced multi-hop wireless mesh network. Propagation can employ routing or flooding.

In computer science and telecommunications, wireless sensor networks are an active research area supporting many workshops and conferences, including International Workshop on Embedded Networked Sensors (EmNetS), IPSN, SenSys, MobiCom and EWSN. As of 2010, wireless sensor networks had deployed approximately 120 million remote units worldwide.

## 2.6 Convolutional LSTM (Convo LSTM)

Temporal information prediction introduced from Long Short-Term Memory (LSTM) Network. The idea represents the sequence data in 1-D space and predicting the future element based on the stacked LSTM layers. To carry the required information which can hold the sequential information designed are important. [2]

Convolutional LSTM is a type of recurrent neural network for spatio-temporal prediction that has convolutional structures in both the input-to-state and state-to-state transitions. The Convolutional LSTM determines the future state of a certain cell in the grid by the inputs and past states of its local neighbours. Data collected over successive periods of time are characterised as a Time Series. In such

cases, an interesting approach is to use a model based on LSTM (Long Short Term Memory), a Recurrent Neural Network architecture. In this kind of architecture, the model passes the previous hidden state to the next step of the sequence. Therefore holding information on previous data the network has seen before and using it to make decisions. In other words, the data order is extremely important.

## 2.7 Auto encoder

Auto encoder based on Artificial Neural Networks (ANN) and machine learning unsupervised techniques use in it. Unsupervised techniques perform various feature detection and classification in the techniques. The auto encoder are the feed forward neural networks, where input and output are almost same are shown in the Fig.1. Two main encoder i. e. Convolution encoder and Classical encoder, the convolution encoder are more efficient than the classical encoder.

Many research works have been done on the detection and prevention of diseases in general and skin in particular. But there are some restrictions on the same. Most of the previous researches use only the principal component analysis, i.e. only one sample skin is taken into consideration for comparison, and the same is maintained as reference filter value throughout the whole process of the detection of the disease. Moreover, during the segmentation process, only the texture-based segmentation techniques [4] [5] [6] are deployed during the detection process. This may have produced the desired results, but the accurate prediction of disease detection [7] [8] will not be the same when the same is applied in real-time scenarios. This may lead to diversion and wrong pesticide or medical suggestions to the farmers, thus resulting in chaos. Mostly [9] [10] [15] K Means Clustering is being used for detection purposes. There are many other techniques available, hence it is necessary to make a detailed study by implementing the same and making comparisons on the level of accurate prediction of the said implementation.

Some research works have been implemented in MATLAB [11] [12]. Statically the measures of this detection system that are processed in terms of reactivity, specificity, and accuracy need to be designed and implemented in a much more sophisticated, user-friendly environment, to achieve the desired detection levels.

It is important to recognize the previous research done in regard to this field to be able to correctly advance in the right direction. Plant skin disease detection has been a major research area in which both image processing and deep learning techniques have been widely used for its accurate classification. In this paper, we discuss the most popularly incorporated techniques in literature in the relevant field.

In order to overcome the problem of the above paper, the authors in [3] have proposed various segmentation, feature extraction and classification techniques that identify and detect the type of the disease using the diseased image to conduct classification. The skin image given as input to the system was pre-processed by smoothing it or enhancing the image by performing histogram equalization. To obtain the affected area, different segmentation techniques like K-Means clustering have been proposed. The features were then extracted from the segmented region and calculated using GLCM. After feature

extraction, the diseases can be detected with the help of Artificial Neural Networks (ANN) or Back Propagation Neural Networks. The drawback of segmenting the image using K-Means clustering is that the process proposed was semi-automated as the user has to explicitly select the cluster which contains the diseased part.

The paper [8] describes a method which makes use of the Gabor wavelet transformation technique for the purpose of feature extraction which helps in the disease identification of skin. The extracted features were input to the SVM classifier for training which then determines the type of disease of the infected skin. Resizing of the images, elimination of noise and background removal have been carried out in the pre-processing stage. The paper has made use of Gabor transformation to identify the textual patterns of the affected skin and extract appropriate features. Disease classification was carried out using SVM with different kernel functions and performance has been evaluated using cross-validation technique. An accuracy of 99.5% has been shown to have achieved according to the experimental results of the system proposed. The main disadvantage of using Gabor transformation for feature extraction is that it is computationally intensive.

In [9], the authors have used a simple approach for the classification of the diseased skin into various classes namely Acne and Rosacea, Actinic Keratosis, Atopic Dermatitis, and Bullous Disease. A dataset of 383 images which have been captured using a digital camera has been used for the purpose of implementation. Otsu's method for image segmentation has been applied on the dataset. Colour features have been obtained using the RGB colour components while shape features have been obtained using region props function and texture features have been obtained from GLCM. All the extracted features have been combined to form a feature extraction module. Supervised learning techniques have been used for classification by training the decision tree classifier. Though the accuracy is high, decision tree has its own set of disadvantages – over fitting in case of noisy data and the amount of control that the user has over the model is relatively less.

Deep convolutional neural networks have been trained in [6] for the identification of 26 diseases in 14 different skin diseases. The authors make use of the standard Alex Net [4] and Google Net [10] architectures for this purpose. A public repository which contains 54,306 images of both diseased leaves and healthy plant leaves has been used for this purpose. The dataset has been created by collecting the images of the plant leaves in a controlled environment.

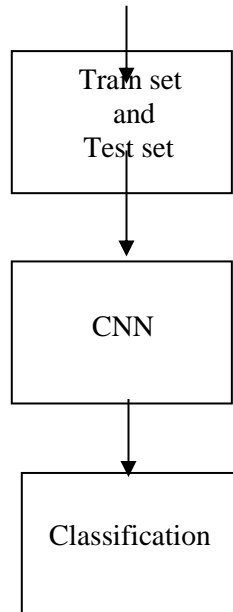
The authors have conducted a performance analysis on both these architectures by carrying out the model training in two ways. It is performed from scratch in the first case and by using transfer learning in the second. Transfer learning corresponds to the process of adapting pre-trained weights obtained by training models on the ImageNet dataset. The model implementation has been carried out using the Caffe framework giving an accuracy of 99%. This portrays the feasibility of this approach. However, on testing the trained model against a set of sample test images obtained from online public data sources which are quite different from the train set, the model accuracy falls to 31.4%. This is a common problem faced in neural networks owed to the train and test sets belonging to different distributions.



### Chapter 3 METHODOLOGY

#### 3.1 CNN Model

The block diagram of method which has been proposed for skin disease classification is presented in Figure 3.1. The method involves various phases such as Data Acquisition, Data pre-processing and classification.



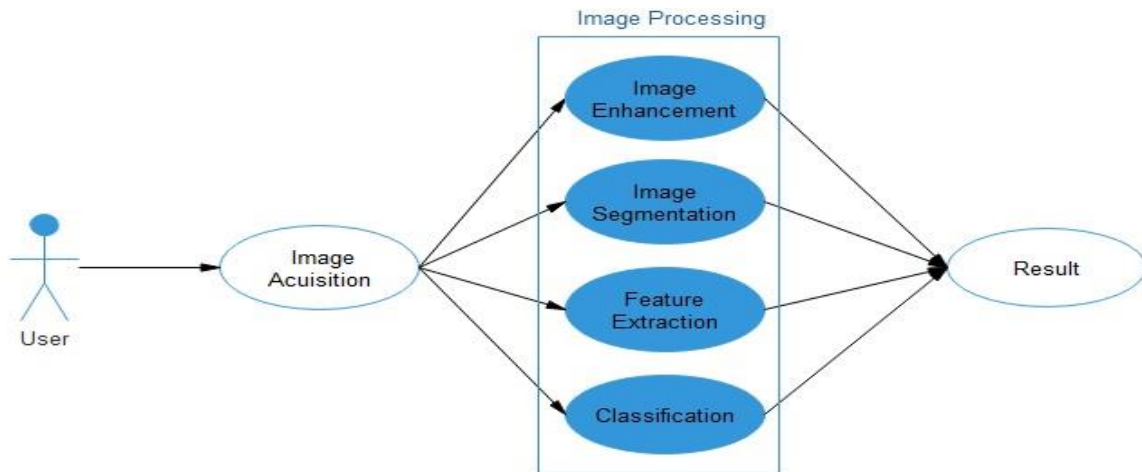
**Figure 3.1: Block diagram of proposed method**

The main component of the proposed model is CNN. The different layers which are used in the proposed model are given in Figure 3.2. The use case diagram for the proposed method is shown in Figure 3.3

#### 3.2 System Design

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 62, 62, 32)	896
max_pooling2d (MaxPooling2D)	(None, 31, 31, 32)	0
dropout (Dropout)	(None, 31, 31, 32)	0
conv2d_1 (Conv2D)	(None, 29, 29, 64)	18496
max_pooling2d_1 (MaxPooling2)	(None, 14, 14, 64)	0
dropout_1 (Dropout)	(None, 14, 14, 64)	0
conv2d_2 (Conv2D)	(None, 12, 12, 128)	73856
max_pooling2d_2 (MaxPooling2)	(None, 6, 6, 128)	0
dropout_2 (Dropout)	(None, 6, 6, 128)	0
flatten (Flatten)	(None, 4608)	0
dense (Dense)	(None, 64)	294976
dense_1 (Dense)	(None, 128)	8320
dense_2 (Dense)	(None, 64)	8256
dense_3 (Dense)	(None, 10)	650
Total params: 405,450		
Trainable params: 405,450		
Non-trainable params: 0		

**Figure 3.2: System design**



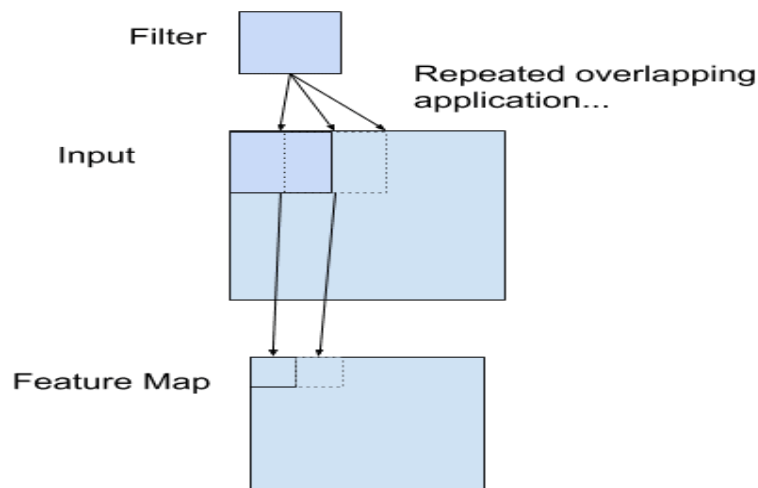
**Figure 3.3 Use case diagram**

### 3.3 Convolutional Layer

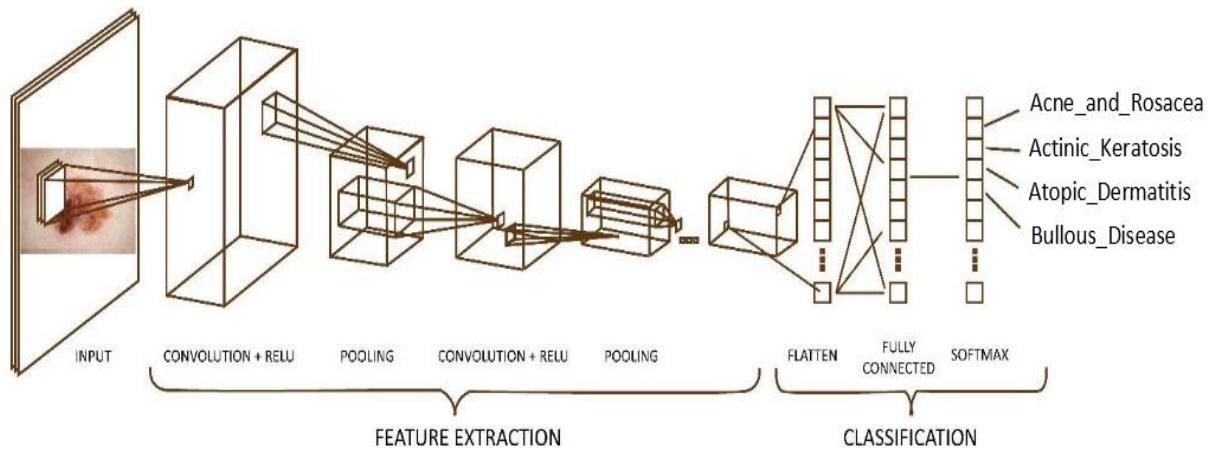
A 2-D convolutional layer applies sliding convolutional filters to 2-D input. The layer convolves the input by moving the filters along the input vertically and horizontally and computing the dot product of the weights and the input, and then adding a bias term.

The dimensions that the layer convolves over depends on the layer input as shown in Figure 3.4:

- For 2-D image input (data with four dimensions corresponding to pixels in two spatial dimensions, the channels, and the observations), the layer convolves over the spatial dimensions.
- For 2-D image sequence input (data with five dimensions corresponding to the pixels in two spatial dimensions, the channels, the observations, and the time steps), the layer convolves over the two spatial dimensions.
- For 1-D image sequence input (data with four dimensions corresponding to the pixels in one spatial dimension, the channels, the observations, and the time steps), the layer convolves over the spatial and time dimensions.



**Figure 3.4: Convolution process**



**Figure 3.5: Convolution Neural Network**

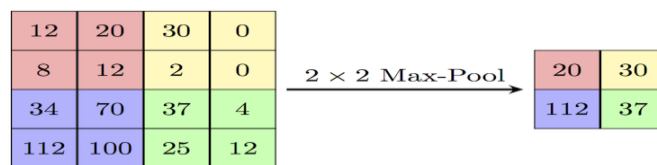
### 3.4 Feature Map

Feature Map is also called as Activation map. Once the filters are extracted from the Image. And these filters are small sections of the image which will be having different features. Number of filters used on the input should be the same number amount of feature maps. So an input image with 6 filters will have 6 feature maps.

For example, Consider a  $32 \times 32$  image, sliding the  $5 \times 5$  receptive field across the learning/input image with a stride width of 1 will result in a feature map of  $28 \times 28$  output values. Convolutional Neural Networks look for "features" such as straight lines, or cats. As such whenever you spot those features-they get reported to the feature map. Each feature map is looking for something else. One feature map could be looking for straight lines, the other for curves. The feature maps also look for their features in different locations.

### 3.5 Max Pooling

Max Pooling as shown in Figure 3.6 is a pooling operation that calculates the maximum value for patches of a feature map, and uses it to create a down sampled (pooled) feature map. It is usually used after a convolutional layer. It adds a small amount of translation invariance - meaning translating the image by a small amount does not significantly affect the values of most pooled outputs.



**Figure 3.6: Max pooling**

Convolutional layers in a convolutional neural network systematically apply learned filters to input images in order to create feature maps that summarize the presence of those features in the input. Convolutional layers prove very effective, and stacking convolutional layers in deep models allows layers close to the input to learn low-level features (e.g. lines)

and layers deeper in the model to learn high-order or more abstract features, like shapes or specific objects.

A limitation of the feature map output of convolutional layers is that they record the precise position of features in the input. This means that small movements in the position of the feature in the input image will result in a different feature map. This can happen with re-cropping, rotation, shifting, and other minor changes to the input image. A common approach to addressing this problem from signal processing is called down sampling. This is where a lower resolution version of an input signal is created that still contains the large or important structural elements, without the fine detail that may not be as useful to the task.

Down sampling can be achieved with convolutional layers by changing the stride of the convolution across the image. A more robust and common approach is to use a pooling layer. A pooling layer is a new layer added after the convolutional layer. Specifically, after a nonlinearity (e.g. RELU) has been applied to the feature maps output by a convolutional layer; for example the layers in a model may look as follows:

- Input Image
- Convolutional Layer
- Nonlinearity
- Pooling Layer

The addition of a pooling layer after the convolutional layer is a common pattern used for ordering layers within a convolutional neural network that may be repeated one or more times in a given model. The pooling layer operates upon each feature map separately to create a new set of the same number of pooled feature maps.

Pooling involves selecting a pooling operation, much like a filter to be applied to feature maps. The size of the pooling operation or filter is smaller than the size of the feature map; specifically, it is almost always  $2 \times 2$  pixels applied with a stride of 2 pixels.

This means that the pooling layer will always reduce the size of each feature map by a factor of 2, e.g. each dimension is halved, reducing the number of pixels or values in each feature map to one quarter the size. For example, a pooling layer applied to a feature map of  $6 \times 6$  (36 pixels) will result in an output pooled feature map of  $3 \times 3$  (9 pixels). The pooling operation is specified, rather than learned. Two common functions used in the pooling operation are:

- Average Pooling: Calculate the average value for each patch on the feature map.
- Maximum Pooling (or Max Pooling): Calculate the maximum value for each patch of the feature map.

The result of using a pooling layer and creating down sampled or pooled feature maps is a summarized version of the features detected in the input. They are useful as small changes in the location of the feature in the input detected by the convolutional layer will result in a pooled feature map

with the feature in the same location. This capability added by pooling is called the model’s invariance to local translation.

### 3.6 Dropout

Dropout is implemented per-layer in a neural network. It can be used with most types of layers, such as dense fully connected layers, convolutional layers, and recurrent layers such as the long short-term memory network layer. Dropout may be implemented on any or all hidden layers in the network as well as the visible or input layer. It is not used on the output layer.

A new hyper-parameter is introduced that specifies the probability at which outputs of the layer are dropped out, or inversely, the probability at which outputs of the layer are retained. The interpretation is an implementation detail that can differ from paper to code library. A common value is a probability of 0.5 for retaining the output of each node in a hidden layer and a value close to 1.0, such as 0.8, for retaining inputs from the visible layer.

The default interpretation of the dropout hyper parameter is the probability of training a given node in a layer, where 1.0 means no dropout, and 0.0 means no outputs from the layer. A good value for dropout in a hidden layer is between 0.5 and 0.8. Input layers use a larger dropout rate, such as of 0.8.

### 3.7 Flattening

Flattening as shown in Figure 3.7 is used to convert all the resultant 2-Dimensional arrays from pooled feature maps into a single long continuous linear vector.

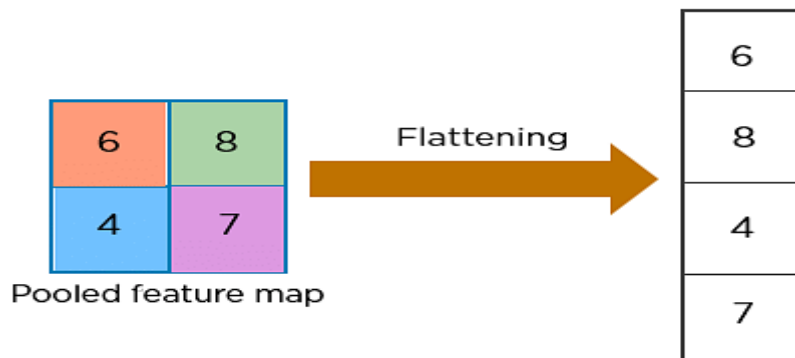


Figure 3.7: Flattening

The reason why we transform the pooled feature map into a one-dimensional vector is because this vector will now be fed into an artificial neural network. Said differently, this vector will now become the input layer of an artificial neural network that will be chained onto the convolutional neural network we've been building so far in this course.

### 3.8 Dense Layer

In any neural network, a dense layer is a layer that is deeply connected with its preceding layer which means the neurons of the layer are connected to every neuron of its preceding layer. This layer is the most commonly used layer in artificial neural network networks. The dense layer’s neuron in a model receives output from every neuron of its preceding layer, where neurons of the dense layer



perform matrix-vector multiplication. Matrix vector multiplication is a procedure where the row vector of the output from the preceding layers is equal to the column vector of the dense layer. The general rule of matrix-vector multiplication is that the row vector must have as many columns like the column vector.

### 3.9 RELU Layer

RELU as shown in Figure 3.8 stands for the rectified linear unit. Once the feature maps are extracted, the next step is to move them to a RELU layer. RELU performs an element-wise operation and sets all the negative pixels to 0. It introduces non-linearity to the network, and the generated output is a rectified feature map. Below is the graph of a RELU function:

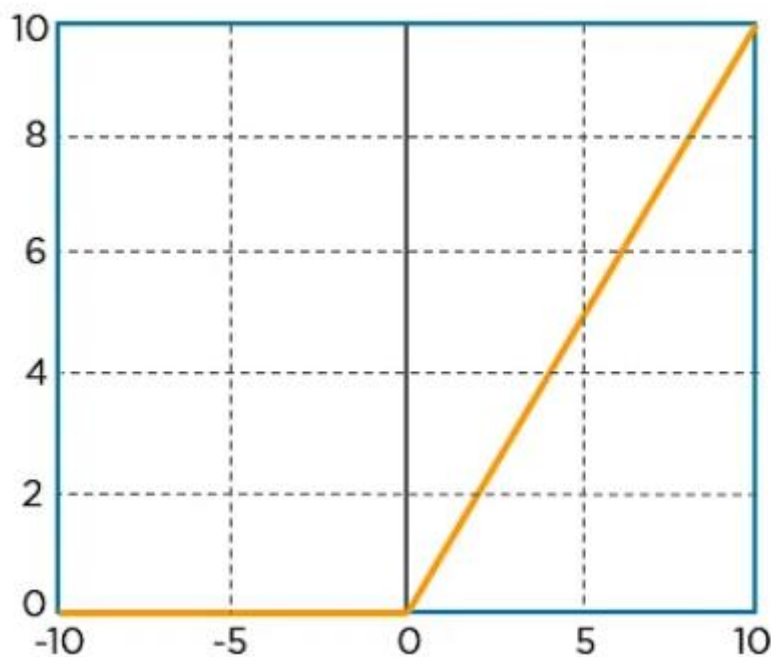


Figure 3.8 RELU activation function

The Rectified Linear Unit is the most commonly used activation function in deep learning models. The function returns 0 if it receives any negative input, but for any positive value  $x$  it returns that value back. So it can be written as  $f(x)=\max(0,x)$ . It's surprising that such a simple function (and one composed of two linear pieces) can allow your model to account for non-linearity's and interactions so well. But the RELU function works great in most applications, and it is very widely used as a result.

### 3.10 Fully Connected Layer

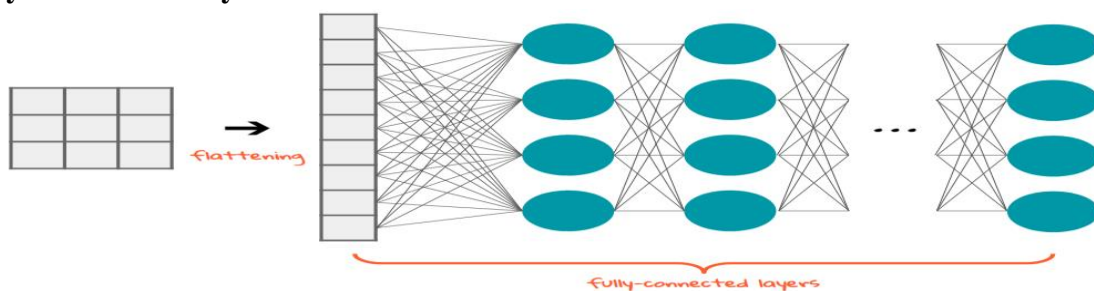


Figure 3.9: Fully connected layer

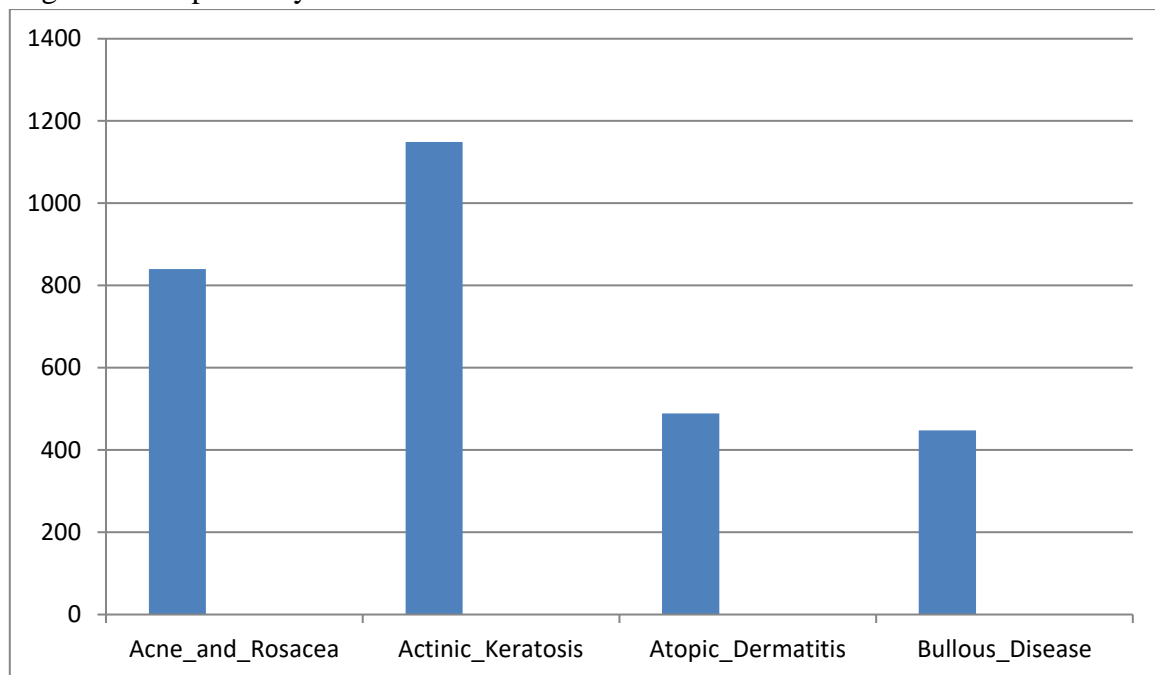
As you can likely infer from the last section, the full connection step involves chaining an artificial neural network onto our existing convolutional neural network. The fully connected layer is as shown in Figure 3.9. The reason this is called the full connection step is because the hidden layer of the artificial neural network is replaced by a specific type of hidden layer called a fully connected layer. As its name implies, a fully connected layer's neurons are connected to all of the neurons in the next layer.

The purpose of the fully connected layer in a convolutional neural network is to detect certain features in an image. More specifically, each neuron in the fully connected layer corresponds to a specific feature that might be present in an image. The value that the neuron passes on to the next layer represents the probability that the feature is contained in the image. The end of the artificial neural network coincides with the end of the convolutional neural network. Said differently, the artificial neural network at the end of a CNN predicts what's contained in the image that the CNN is attempting to recognize.

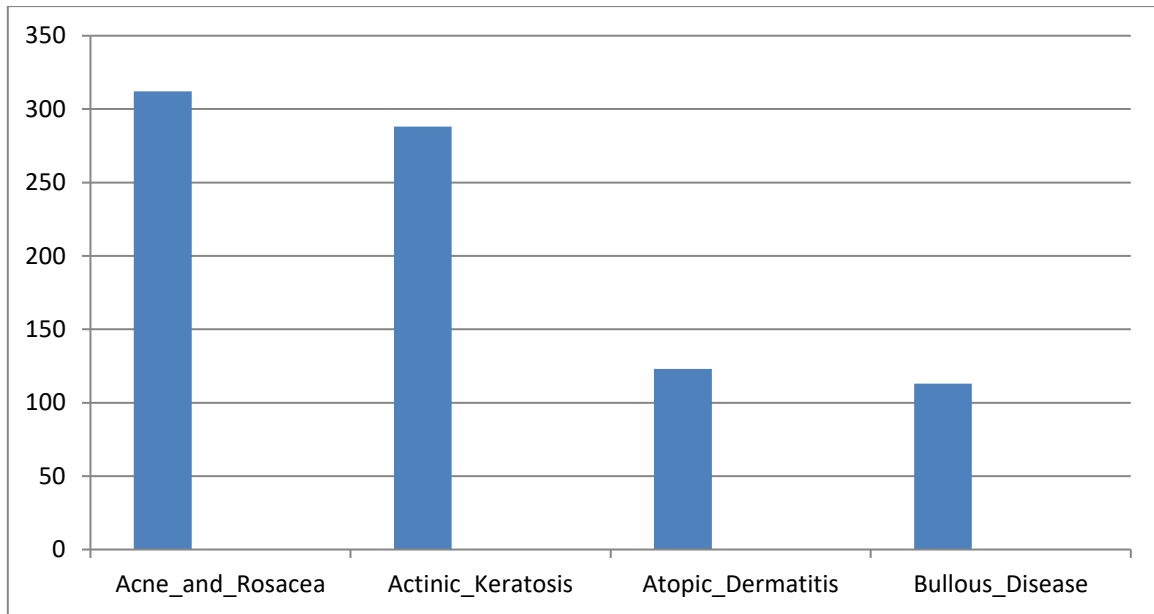
## Chapter 4

### RESULTS AND DISCUSSION

The number of sample images used for training and testing in each category is shown in Figure 4.1 and Figure 4.2 respectively.



**Figure 4.1: Training images**



**Figure 4.2: Testing images**

Deep learning neural networks are trained using the stochastic gradient descent optimization algorithm. As part of the optimization algorithm, the error for the current state of the model must be estimated repeatedly. This requires the choice of an error function, conventionally called a loss function, that can be used to estimate the loss of the model so that the weights can be updated to reduce the loss on the next evaluation.

Neural network models learn a mapping from inputs to outputs from examples and the choice of loss function must match the framing of the specific predictive modelling problem, such as classification or regression. Further, the configuration of the output layer must also be appropriate for the chosen loss function.

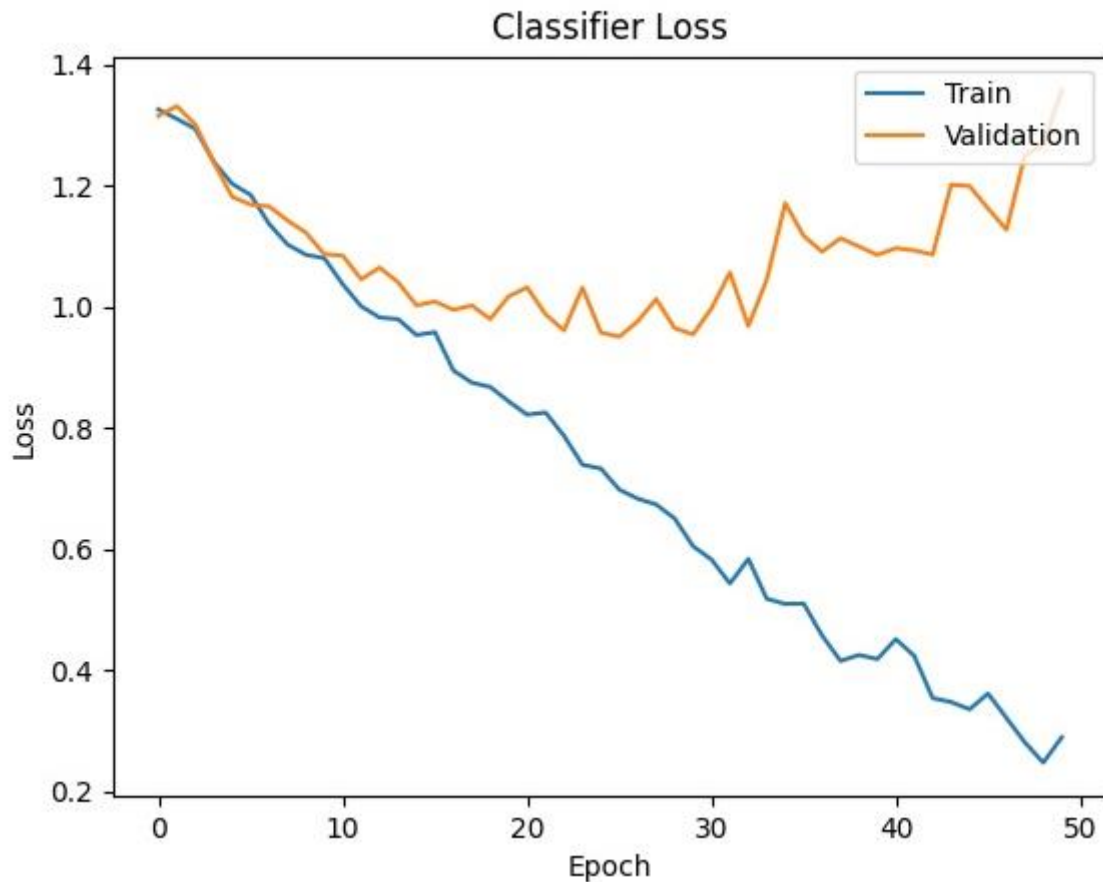
#### 4.1 Binary Cross Entropy Loss

This is the most common loss function used for classification problems that have two classes. The word “entropy”, seemingly out-of-place, has a statistical interpretation. Entropy is the measure of randomness in the information being processed, and cross entropy is a measure of the difference of the randomness between two random variables.

If the divergence of the predicted probability from the actual label increases, the cross-entropy loss increases. Going by this, predicting a probability of .011 when the actual observation label is 1 would result in a high loss value. In an ideal situation, a “perfect” model would have a log loss of 0 as given in Eq. 1. Looking at the loss function as shown in Fig. 14 would make things even clearer.

$$J = - \sum_{i=1}^N y_i \log(h_{\theta}(x_i)) + (1 - y_i) \log(1 - h_{\theta}(x_i)) \tag{1}$$

Where  $y_i$  is the true label and  $h_{\theta}(x_i)$  is the predicted value post hypothesis. Since binary classification means the classes take either 0 or 1, if  $y_i = 0$ , that term ceases to exist and if  $y_i = 1$ , the  $(1 - y_i)$  term becomes 0. The loss during training and validation through epochs is as shown in Figure 4.3.



**Figure 4.3: Loss curve**

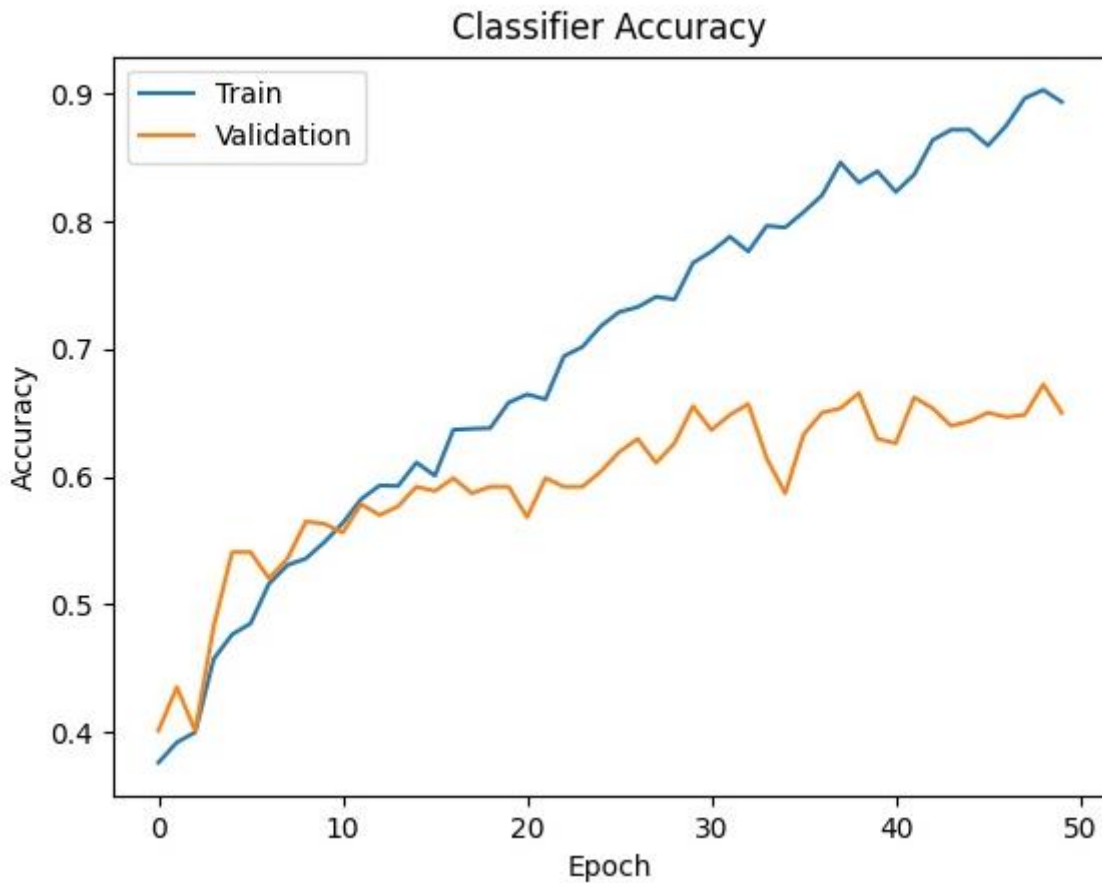
## 4.2 Classification Accuracy

Classification accuracy is a metric that summarizes the performance of a classification model as the number of correct predictions divided by the total number of predictions. It is easy to calculate and intuitive to understand, making it the most common metric used for evaluating classifier models. This intuition breaks down when the distribution of examples to classes is severely skewed.

Intuitions developed by practitioners on balanced datasets, such as 99 per cent representing a skill full model, can be incorrect and dangerously misleading on imbalanced classification predictive modelling problems. In this tutorial, you will discover the failure of classification accuracy for imbalanced classification problems.

After completing this tutorial, you will know: Accuracy and error rate are the de facto standard metrics for summarizing the performance of classification models. Classification accuracy fails on classification problems with a skewed class distribution because of the intuitions developed by practitioners on datasets with an equal class distribution.

Intuition for the failure of accuracy for skewed class distributions with a worked example. The classification accuracy through epochs is as shown in Figure 4.4.



**Figure 4.4: Classification accuracy**

Classification accuracy involves first using a classification model to make a prediction for each example in a test dataset. The predictions are then compared to the known labels for those examples in the test set. Accuracy is then calculated as the proportion of examples in the test set that were predicted correctly, divided by all predictions that were made on the test set.

$$\text{Accuracy} = \text{Correct Predictions} / \text{Total Predict}$$

Conversely, the error rate can be calculated as the total number of incorrect predictions made on the test set divided by all predictions made on the test set.

$$\text{Error Rate} = \text{Incorrect Predictions} / \text{Total Predictions}$$

The accuracy and error rate are complements of each other, meaning that we can always calculate one from the other. For example:

$$\text{Accuracy} = 1 - \text{Error Rate}$$

$$\text{Error Rate} = 1 - \text{Accuracy}$$

Another valuable way to think about accuracy is in terms of the confusion matrix.

The classification accuracy can be calculated from this confusion matrix as the sum of correct cells in the table (true positives and true negatives) divided by all cells in the table. The accuracy is calculated using Eq. (2)

$$\text{Accuracy} = (\text{TP} + \text{TN}) / (\text{TP} + \text{FN} + \text{FP} + \text{TN}) \tag{2}$$

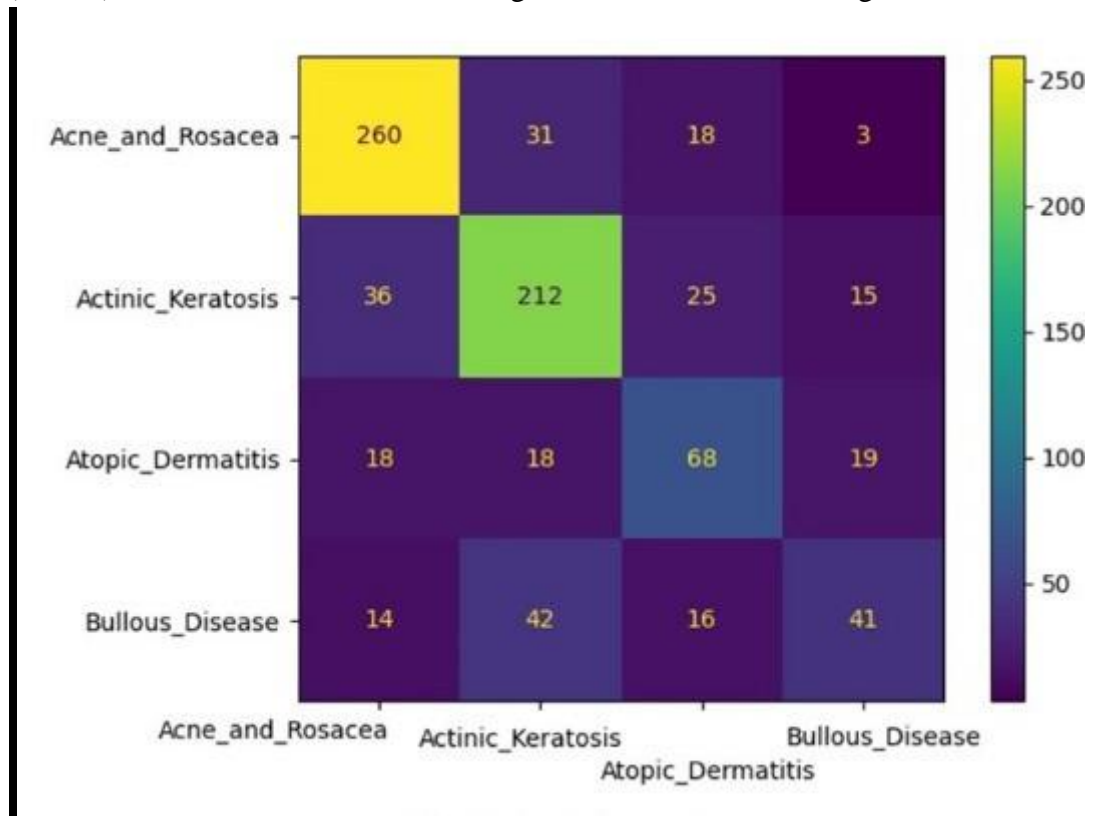
Similarly, the error rate can also be calculated from the confusion matrix as the sum of incorrect cells of the table (false positives and false negatives) divided by all cells of the table. The error rate is calculated using the Eq. (3).



$$\text{Error Rate} = (\text{FP} + \text{FN}) / (\text{TP} + \text{FN} + \text{FP} + \text{TN}) \tag{3}$$

A confusion matrix is a summary of the predictions made by a classification model organized into a table by class. Each row of the table indicates the actual class and each column represents the predicted class. A value in the cell is a count of the number of predictions made for a class that are actually for a given class. The cells on the diagonal represent correct predictions, where a predicted and expected class align.

The confusion matrix provides more insight into not only the accuracy of a predictive model, but also which classes are being predicted correctly, which incorrectly, and what type of errors are being made. The simplest confusion matrix is for a two-class classification problem, with negative (class 0) and positive (class 1) classes. The confusion matrix generated is as shown in Figure 4.5.



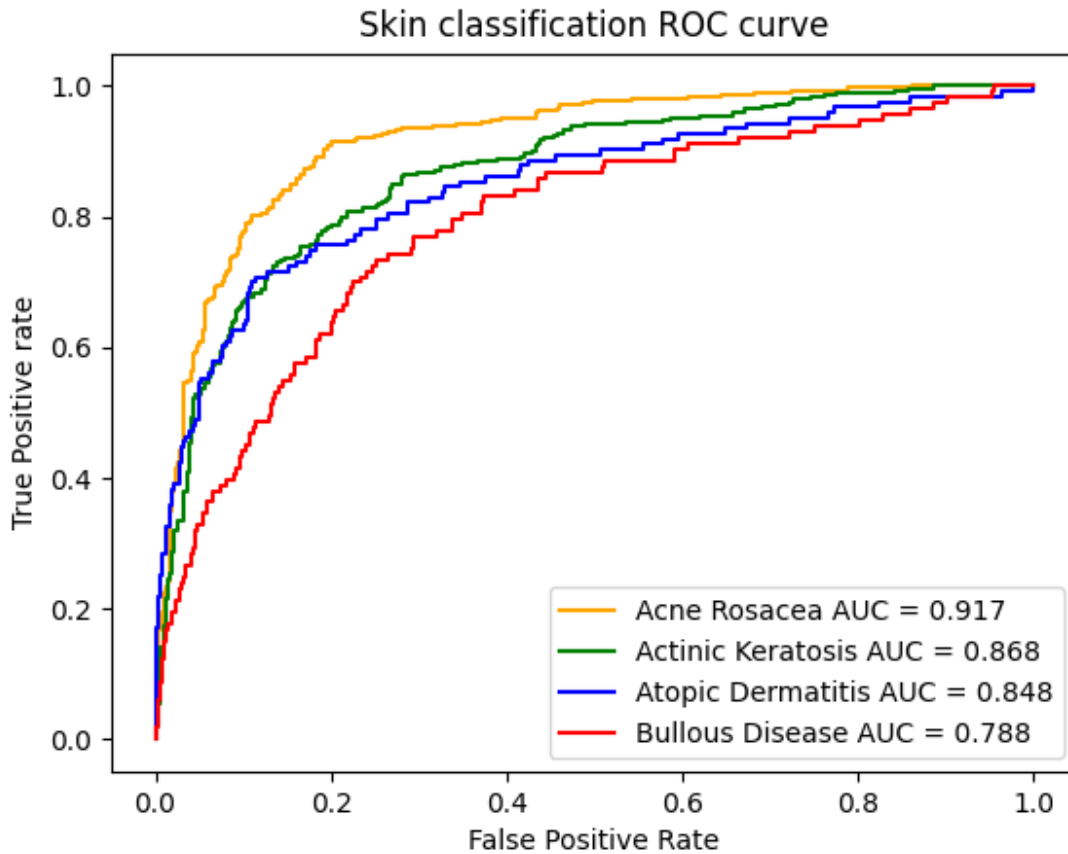
**Figure 4.5: Confusion matrix**

A confusion matrix is a matrix that summarizes the performance of a machine learning model on a set of test data. It is often used to measure the performance of classification models, which aim to predict a categorical label for each input instance. The matrix displays the number of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) produced by the model on the test data.

For binary classification, the matrix will be of a 2X2 table, For multi-class classification, the matrix shape will be equal to the number of classes i.e for n classes it will be nXn.

A true positive is an outcome where the model correctly predicts the positive class. Similarly, a true negative is an outcome where the model correctly predicts the negative class. A false positive is an

outcome where the model incorrectly predicts the positive class. A false negative is an outcome where the model incorrectly predicts the negative class.



**Figure 4.6: ROC Curve**

ROC curves as shown in Figure 4.6 typically feature true positive rate (TPR) on the Y axis, and false positive rate (FPR) on the X axis. This means that the top left corner of the plot is the “ideal” point - a FPR of zero, and a TPR of one. This is not very realistic, but it does mean that a larger area under the curve (AUC) is usually better. The “steepness” of ROC curves is also important, since it is ideal to maximize the TPR while minimizing the FPR.

ROC curves are typically used in binary classification, where the TPR and FPR can be defined unambiguously. In the case of multiclass classification, a notion of TPR or FPR is obtained only after binarizing the output. This can be done in 2 different ways:

1. The One-vs-Rest scheme compares each class against all the others (assumed as one).
2. The One-vs-One scheme compares every unique pair-wise combination of classes.

The True Positive Rate (TPR) and False Positive Rate (FPR) are calculated using the Eq. (4) and Eq. (5) respectively.

$$TPR = \frac{\sum_c TP_c}{\sum_c (TP_c + FN_c)} \quad (4)$$

$$FPR = \frac{\sum_c FP_c}{\sum_c (FP_c + TN_c)} \quad (5)$$

False positive rate (FPR) is a measure of accuracy for a test: be it a medical diagnostic test, a machine learning model, or something else. In technical terms, the false positive rate is defined as the probability of falsely rejecting the null hypothesis.

The false positive rate is calculated as  $FP/FP+TN$ , where FP is the number of false positives and TN is the number of true negatives (FP+TN being the total number of negatives). It's the probability that a false alarm will be raised: that a positive result will be given when the true value is negative.

There are many other possible measures of test accuracy and error rate. Here is a short rundown of the most common ones:

1. The false negative rate – also called the miss rate – is the probability that a true positive will be missed by the test. It's calculated as  $FN/FN+TP$ , where FN is the number of false negatives and TP is the number of true positives (FN+TP being the total number of positives).
2. The true positive rate (TPR, also called sensitivity) is calculated as  $TP/TP+FN$ . TPR is the probability that an actual positive will test positive.
3. The true negative rate (also called specificity), which is the probability that an actual negative will test negative. It is calculated as  $TN/TN+FP$ .

If you're on the patient side of a medical test being analysed like this, you may care a bit more about two additional metrics: positive predictive value and negative predictive value. Positive predictive value is the likelihood that, if you have gotten a positive test result, you actually have the disease. It's calculated as  $TP/TP+FP$ . Conversely, negative predictive value is the likelihood that, if you have gotten a negative test result, you actually don't have the disease.

## Chapter 5

### CONCLUSION AND FUTURE SCOPE

Skin classification is an important research that should be looked into and it has more research and development resources because its development and improvement can have a far-reaching effect in medical field. The maximum accuracy obtained is 90%. The deep learning models are usually built using neural networks. The proposed approach uses different CNN layers outputs along with fully connected layers for creating a model to classify skin disease. The max pooling extracts and resize the image more better than average pooling method.

In future, the proposed model need to be tested with different datasets of skin diseases. Also, the efficiency of proposed model need to be tested for different kinds of diseases.

## References

1. M. E., J. Newton, S.-B. S., S.-B. S., R. T. and F. A., "The quality of life in acne: a comparison with general medical conditions using generic questionnaires.," *British Journal of Dermatology*, pp. 672-676, 1999.
2. H. J., S. R., D. F., T. M., B. E. and L. L., "Suicidal ideation, mental health problems, and social impairment are increased in adolescents with acne: a population-based study.," *Journal of Investigative Dermatology*, pp. 363-370, 2011.
3. R. E. and K. A., "New photographic techniques for clinical evaluation of acne.," *Journal of the European Academy of Dermatology and Venereology*, pp. 13-18, 2001.
4. D. B. and P. F., "Epidemiology of acne," *Dermatology*, pp. 7-10.
5. J. Tan, "Current measures for the evaluation of acne severity," *Expert Review of Dermatology*, pp. 595-603, 2008.
6. C. L., L. S., H. M., H. K. and E. S., "Acne vulgaris: a disease of Western civilization," *Archives of dermatology*, pp. 1584-1590, 2002.
7. G. H., "From new findings in acne pathogenesis to new approaches in treatment," *J Eur Acad Dermatol Venereol*, pp. 1-7, 2015.
8. J. Huizen, "Acne types in pictures: Explanations and treatments," *Medical News Today*, 29 June 2018. [Online]. Available: <https://www.medicalnewstoday.com/articles/322322.php#causes>. [Accessed 1 Feb 2020].
9. C. K. and .. M. A. K. E. B. Stark, "A Literature Review on Medicine Recommender Systems," (*IJACSA*) *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 8, 2019.
10. Brito, M. d., Anna, I. P., Galindo, J. C., Rosendo, L. H. And &. Santos, "Evaluation of clinical adverse effects and laboratory alterations in patients with acne vulgaris treated with oral isotretinoin.," *An Bras Dermatol*, pp. 331-336, 2010.
11. A. M., V. f., V. M., H. K., Z. H., K. W. and M. N., "Automated facial acne assessment from smartphone images," San Francisco, California, United States, 2018.
12. DermNet NZ, "Sensitive skin | DermNet NZ," [Online]. Available: <https://dermnetnz.org/topics/sensitive-skin/>. [Accessed 1 Feb 2020].
13. N. Alamdari, K. Tavakolian, M. Alhashim and R. Fazel- Rezai, "Detection and classification of acne lesions in acne patients: A mobile application," in 2016 IEEE International Conference on Electro Information Technology (EIT), 2016.
14. S. Akyeramfo-Sam, A. A. Philip, D. Yeboah, N. C. Nartey and I. K. Nti, ""A WebBased Skin Disease Diagnosis Using Convolutional Neural Networks," *International Journal of Information Technology and Computer Science*, 2019.
15. X. Wu, N. Wen, J. Liang, Y.-K. Lai, D. She, M.-M. Cheng and J. Yang, "Joint Acne Image Grading and Counting via Label Distribution Learning," in *International Conference on Computer Vision (ICCV)*, IEEE Xplore, 2019.
16. K. He, X. Zhang, S. Ren and j. Sun, "Deep residual learning for image recognition," *CVPR*, 2016.